



Escuela
Politécnica
Superior

Previsión de stock con técnicas de Inteligencia Artificial para distribuidoras de alimentación y bebidas



Grado en Ingeniería Informática

Trabajo Fin de Grado

Autor:

Carlos Garrido Marín

Tutor/es:

Miguel Ángel Cazorla Quevedo

Junio 2021



Universitat d'Alacant
Universidad de Alicante

Previsión de stock con técnicas de Inteligencia Artificial para distribuidoras de alimentación y bebidas

Autor

Carlos Garrido Marín

Tutor/es

Miguel Ángel Cazorla Quevedo

CIENCIA DE LA COMPUTACION E INT. ARTIF.



Grado en Ingeniería Informática



Escuela
Politécnica
Superior



Universitat d'Alacant
Universidad de Alicante

ALICANTE, Junio 2021

Agradecimientos

Este trabajo no habría sido posible sin el apoyo y el estímulo de mis amigos y familia. Agradecer a: mi padre, por haberme enseñado a ser constante, a no rendirse en los momentos más duros y que el día parezca tener más de 24 horas; a mi madre, por todo el apoyo y cariño que me ha dado desde la distancia desde el momento que partí a Alicante; a mi amigo Juan, por todas esas noches en vela de estudio y por haber leído 20 veces como mínimo este trabajo en busca de errores; a mi pareja, por apoyarme en todo lo que he hecho y confiar en mí en todas las locuras a las que me he aventurado a lo largo de estos 4 años.

También agradecer a mi mentor en las empresas de distribución de alimentación y bebidas, César, que más que un jefe, lo puedo considerar un amigo. Por enseñarme todo acerca de estas empresas de manera amena y en base a su experiencia, por hacerme ver lo increíble que es este sector hasta el punto de haber partido en la aventura de hacer este trabajo orientado a este sector. Mil gracias.

No puedo terminar sin agradecer de todo corazón a mi tutor Miguel, un gran profesional, pero aún más importante, una excelente persona. Gracias por haberme orientado en este trabajo, por haber estado tan atento y por tener tanta paciencia.

Es a todos ellos a quien dedico este trabajo.

Considero más valiente al que conquista sus deseos, que al que conquista a sus enemigo, ya que la victoria más dura es la victoria sobre uno mismo

Aristóteles.

Índice general

1	Introducción	1
1.1	Sector de la alimentación y bebidas	1
1.2	Motivación	1
1.3	Objetivos	2
2	Estado del arte	3
2.1	Pronóstico ingenuo	3
2.2	Media móvil	5
2.3	Suavizado exponencial	6
2.4	Suavizado exponencial doble	10
3	Metodología	13
3.1	Aprendizaje Automático	13
3.1.1	Árboles de decisión	13
3.1.2	Bosques aleatorios y extremadamente aleatorios	15
3.1.3	AdaBoost	17
3.1.4	Gradient Boosting y Extreme Gradient Boosting	17
3.2	Aprendizaje Profundo	18
4	Desarrollo	21
4.1	Predicción de stock	21
4.1.1	Factores	22
4.1.1.1	Factores Internos	22
4.1.1.1.1	Promociones de productos	22
4.1.1.1.2	Rapples sobre ventas	22
4.1.1.1.3	Precio	23
4.1.1.1.4	Depósitos	24
4.1.1.2	Factores Externos	24
4.1.1.2.1	Familia del producto y estacionalidades	24
4.1.1.2.2	Clima	24
4.1.1.2.3	IPC	25
4.2	Gestión de los datos	26
4.2.1	Obtención de los datos	26
4.2.2	Procesamiento y estructura	26
4.3	Ajuste de parámetros en los modelos	29
4.3.1	Árbol de regresión	29
4.3.2	Bosques extramadamente aleatorios	29
4.3.3	Adaboost con clasificador base de árboles de regresión	30
4.3.4	Extreme Gradient Boosting	31

4.3.5	Perceptrón multicapa	31
5	Experimentación y resultados	33
5.1	Resultados de los modelos	35
5.1.1	Árbol de regresión	35
5.1.2	Bosque extremadamente aleatorio	35
5.1.3	Adaboost con árboles de regresión como clasificadores base	35
5.1.4	Extreme Gradient Boosting	36
5.1.5	Perceptrón Multicapa	37
5.2	Análisis de los resultados obtenidos	44
6	Conclusiones y trabajos futuros	49
	Bibliografía	51

Índice de figuras

2.1	Gráfica de la demanda histórica del producto expuesta en la Tabla 2.1.	4
2.2	Gráfica comparativa de la demanda histórica de la Tabla 2.2 y la predicción por pronóstico ingenuo.	5
2.3	Gráfica de la demanda histórica del producto expuesta en la Tabla 2.3.	6
2.4	Gráfica comparativa de la demanda histórica y la predicción de la demanda resultante del método de la media móvil.	7
2.5	Algoritmo de suavizado exponencial simple para la previsión de la demanda (Vandeput, 2021).	8
2.6	Gráfica de la demanda histórica del producto con los datos expuestos en la Tabla 2.5.	8
2.7	Gráfica comparativa de la demanda histórica y la predicción resultante del método de suavizado exponencial simple con los datos de la Tabla 2.5.	9
2.8	Gráfica de la demanda definida en la Tabla 2.6.	11
2.9	Gráfica comparativa de la demanda histórica y la predicción resultante del método de suavizado exponencial doble exponiendo los resultados de la Tabla 2.7.	12
3.1	Árbol de decisión.	14
3.2	Árbol de regresión y sus particiones en un hiper rectángulo.	15
3.3	Funcionamiento de un bosque aleatorio (Yiu, 2019).	16
3.4	Funcionamiento del algoritmo Adaboost con árboles de decisión (Cheng, 2019).	18
3.5	Diagrama de un perceptrón con cinco señales de entrada (Cartas, s.f.).	19
3.6	Funcionamiento de un perceptrón multicapa	19
4.1	Ejemplo de un rapple con sus correspondientes tramos	23
4.2	Gráfica comparativa de la demanda histórica de un producto en diferentes años.	25
4.3	Ejemplo de dos líneas de venta de un albarán de venta.	27
5.1	Comparación de los valores reales de demanda de un producto con los predichos por el modelo del árbol de regresión para el Escenario 1 B.	36
5.2	Comparación de los valores reales de demanda de un producto con los predichos por el modelo del árbol de regresión para el Escenario 2 B.	37
5.3	Comparación de los valores reales de demanda de un producto con los predichos por el modelo de bosque extremadamente aleatorio para el Escenario 1 B.	38
5.4	Comparación de los valores reales de demanda de un producto con los predichos por el modelo de bosque extremadamente aleatorio para el Escenario 2 B.	38
5.5	Comparación de los valores reales de demanda de un producto con los predichos por el modelo de Adaboost con árboles de regresión como clasificadores base para el Escenario 1 B.	39

5.6	Comparación de los valores reales de demanda de un producto con los predichos por el modelo de Extreme Gradient Boosting para el Escenario 2 B.	40
5.7	Comparación de los valores reales de demanda de un producto con los predichos por el modelo de Extreme Gradient Extreme para el Escenario 1 B.	41
5.8	Comparación de los valores reales de demanda de un producto con los predichos por el modelo de Extreme Gradient Extreme para el Escenario 2 B.	41
5.9	Comparación de los valores reales de demanda de un producto con los predichos por el modelo perceptrón multicapa para el Escenario 1 B.	42
5.10	Comparación de los valores reales de demanda de un producto con los predichos por el modelo perceptrón multicapa para el Escenario 2 B.	43
5.11	Demanda histórica de producto con un promedio de consumo de más de 50. .	45
5.12	Demanda histórica de producto con un promedio de consumo de más de 50. .	47

Índice de tablas

2.1	Demanda del producto en diferentes periodos de tiempo.	4
2.2	Demanda histórica de la Tabla 2.1 y predicción por pronóstico ingenuo. . . .	5
2.3	Demanda de un producto en diferentes periodos de tiempo.	6
2.4	Demanda histórica y predicción del producto por el método de la media móvil.	6
2.5	Demanda del producto en diferentes periodos de tiempo.	8
2.6	Demanda histórica del producto en diferentes periodos de tiempo.	11
2.7	Resultados de aplicar modelo de suavizado exponencial doble a la demanda histórica de la Tabla 2.5.	12
4.1	Consumo diario de cada uno de los productos en su formato por defecto. . . .	28
4.2	Consumo por periodos de cada uno de los productos en su formato por defecto.	28
4.3	Estructura de datos para el entrenamiento de los modelos.	28
4.4	Resultado de la búsqueda aleatoria de hiperparámetros para el modelo de árbol de regresión.	29
4.5	Resultado de la búsqueda aleatoria de hiperparámetros para el modelo de bosques extremadamente aleatorios.	30
4.6	Resultado de la búsqueda aleatoria de hiperparámetros para el modelo de Adaboost con clasificador base de árbol de regresión.	30
4.7	Resultado de la búsqueda aleatoria de hiperparámetros para el modelo Extreme Gradient Bosting.	32
4.8	Resultado de la búsqueda aleatoria de hiperparámetros para el modelo de perceptrón multicapa.	32
5.1	Resultados de los indicadores seleccionados aplicando un árbol de regresión en los diferentes escenarios.	34
5.2	Resultados de los indicadores seleccionados aplicando un árbol de regresión en los diferentes escenarios.	36
5.3	Resultados de los indicadores seleccionados aplicando bosques extremadamente aleatorios en los diferentes escenarios.	37
5.4	Resultados de los indicadores seleccionados aplicando Adaboost con árboles de regresión como clasificadores base en los diferentes escenarios.	39
5.5	Resultados de los indicadores seleccionados aplicando el modelo Extreme Gra- dient Boosting en los diferentes escenarios.	40
5.6	Resultados de los indicadores seleccionados aplicando el modelo perceptrón multicapa en los diferentes escenarios.	42
5.7	Entradas y salidas de un producto de con un promedio de consumo de más de 50.	46

1 Introducción

1.1 Sector de la alimentación y bebidas

En España, el sector de la alimentación y bebidas representa una parte importante del PIB, según el INE, abarca un total del 2,6%. En la última década, el crecimiento de este sector ha aumentado hasta 2020, cuando su crecimiento disminuyó debido a la crisis sanitaria.

Dentro de este sector se encuentran las empresas de distribución de alimentación y bebidas que son empresas comerciales dedicadas a la compra de productos de diferentes proveedores para venderlos posteriormente a sus clientes. Un ciclo estándar de negocio en estas empresas consiste en: realizar pedidos a diferentes proveedores de los productos que no se dispone stock, recepción de mercancía, organizar la mercancía en el almacén, recibir un pedido de un cliente, preparar el pedido y entregarlo al cliente.

Con la creciente digitalización y fuerte aumento del volumen de negocio, los procesos dinámicos de estas empresas se han ido haciendo cada vez más complicados de gestionar. Para facilitar su gestión surgieron los programas o softwares ERP especializados en la distribución de alimentación y bebidas. Sin embargo, la necesidad de la gestión de nuevos tipos de datos y estructuras de información cada vez más complejas hace que los softwares ERP de distribución de alimentación y bebidas se queden pronto obsoletos ante nuevos retos y problemas.

Uno de los problemas clave a los que se enfrentan los ERP en este sector, y en lo que se centrará el desarrollo de mi trabajo, es el mantenimiento y reducción de inventario o stock de los productos. La optimización y previsión de stock se convierten en ventajas competitivas muy importantes en una empresa de distribución de alimentación y bebidas, ya que así se consigue un abastecimiento óptimo en cuanto a la previsión de la demanda al menor coste posible. Debido a la naturaleza de los productos y tipo de negocio de estas empresas, disponer de un inventario optimizado es de vital importancia ya que un inventario menor o mayor al necesario para abastecer la demanda tiene un gran coste. A la hora de realizar un pedido a un proveedor estas empresas se enfrentan a la cuestión clave de qué productos y cantidad de cada uno de ellos debe pedirse en un momento dado. Para poder responder a esta pregunta y plantear una posible solución al problema debemos tener en cuenta múltiples factores como qué demanda del producto ha habido en anteriores periodos, a qué precio se encontraba el producto cuando se vendió, la naturaleza del producto, etc. Todos estos factores y soluciones actuales a estos problemas se analizarán a lo largo de este trabajo.

1.2 Motivación

Actualmente el sector de la distribución de alimentación y bebidas en España no dispone de herramientas de software con modelos avanzados de previsión de stock. Cabe destacar, no obstante, que en ciertas tareas como, por ejemplo, en la distribución, estas empresas sí utilizan modelos avanzados de optimización de rutas, que permiten a los vehículos reducir

los tiempos de trayecto, ahorrándose costes de combustible y personal. Sin embargo, en la mayoría de los casos la optimización del stock se basa en la experiencia acumulada por los encargados de su gestión o en modelos básicos estadísticos que son incapaces de predecir eficientemente factores externos o desviaciones atípicas.

La optimización de la predicción del stock constituye, por tanto, un problema real de las empresas de distribución de alimentos y bebidas al que los desarrolladores de software del sector aún no han sabido aportar soluciones concluyentes o eficaces. La predicción y gestión correcta del stock se convierte así en una oportunidad para el desarrollo de herramientas de software adecuadas, pues cada vez son más las empresas que buscan conseguir los beneficios aportados por los modelos avanzados de predicción en todos sus ámbitos de negocio.

El creciente interés de las grandes y pequeñas distribuidoras por el uso de modelos de predicción avanzados se pone de manifiesto en un estudio de la consultora especializada en administración estratégica McKinsey. Este estudio concluye que en las empresas que aplican modelos de inteligencia artificial tienen un ahorro en costes de mantenimiento y reducción de stock: el 61% documenta una disminución de los costes y el 53% un aumento de los ingresos (Cam y cols., 2021).

1.3 Objetivos

El objetivo principal de este Trabajo de Fin de Grado es explorar la aplicación de técnicas de inteligencia artificial como alternativa a las técnicas habituales de gestión y optimización de stocks utilizadas por las empresas de distribución de alimentación y bebidas. Para lograr este objetivo, utilizaré datos suministrados por empresas del sector que me permitirán entrenar diferentes modelos y evaluar aquellos que mejor se adaptan a la problemática planteada.

Los objetivos específicos son:

- Presentar el estado del arte de los modelos estadísticos comúnmente utilizados en la optimización del stock y considerar sus inconvenientes.
 - Introducir los modelos de aprendizaje automático y profundo utilizados en este estudio.
 - Realizar un análisis de los factores internos y externos que afectan a la previsión de stock en las empresas del sector.
 - Procesar, estructurar y tratar los datos proporcionados por las empresas teniendo en cuenta los factores examinados previamente.
 - Determinar los hiperparámetros utilizados en los modelos.
 - Analizar los resultados obtenidos por los modelos en diferentes escenarios.
 - Comparar los resultados de los modelos.
 - En relación con las conclusiones obtenidas, examinar las perspectivas futuras de la aplicación de inteligencia artificial a esta problemática.
-

2 Estado del arte

En este apartado haré un breve recorrido histórico sobre los distintos métodos utilizados en la previsión de stock. Estos irán desde los métodos más sencillos basados en simples fórmulas matemáticas que utilizamos en nuestro día a día sin darnos cuenta, a métodos más complejos basados en modelos estadísticos y econométricos. El objetivo es analizar como ha ido avanzando el campo de la predicción orientado hacia la predicción de stock.

Antes de empezar, es conveniente definir el concepto de periodo de demanda que lo veremos de manera reiterada a lo largo de este trabajo. El periodo de demanda es la cantidad de producto que se ha vendido a lo largo de un periodo de tiempo. Dicho de otra manera, el stock mínimo de producto en un periodo de tiempo que se necesita tener en el inventario para abastecer la demanda. Este periodo de tiempo, variará dependiendo del producto como veremos más adelante.

2.1 Pronóstico ingenuo

Empezaremos analizando el método más arcaico de predicción. El pronóstico ingenuo consiste en basar la previsión de un periodo de demanda f_i en función del último periodo de demanda histórica del producto t_{i-1} . Viene dada por la siguiente formulación:

$$f_i = t_{i-1} \quad (2.1)$$

$$i > 1 \quad (2.2)$$

donde:

t_{i-1} Demanda histórica del periodo $i - 1$.

f_i Predicción del periodo i .

El pronóstico ingenuo nos permitirá hacer pronósticos inmediatos o a corto plazo. Este método tiene grandes limitaciones pues no podremos asegurar una previsión adecuada si en el último periodo de medición de demanda se producen valores atípicos cuando la variable a modelar se ve afectada por factores externos de ocurrencia poco probable; en este contexto, debemos introducir el concepto de **ruido** que son esas variaciones en los datos observados. Como veremos más adelante, existen métodos más complejos capaces de “suavizar” este tipo de desviaciones atípicas.

A continuación, vamos exponer un ejemplo para entender el concepto de este método y el concepto del ruido.

Imaginemos que tenemos el histórico de demanda de un producto por unidades que se muestra en la Tabla 2.1 y en la Figura 2.1.

Haciendo un análisis de estos datos podemos ver que el periodo t_7 posee un valor atípico, pero, ¿cómo se reconoce este tipo de valores?. Una posible manera sería calcular los límites

Periodo (i)	1	2	3	4	5	6	7	8	9	10
Demanda (t)	6	4	3	5	7	6	17	3	5	4

Tabla 2.1: Demanda del producto en diferentes periodos de tiempo.

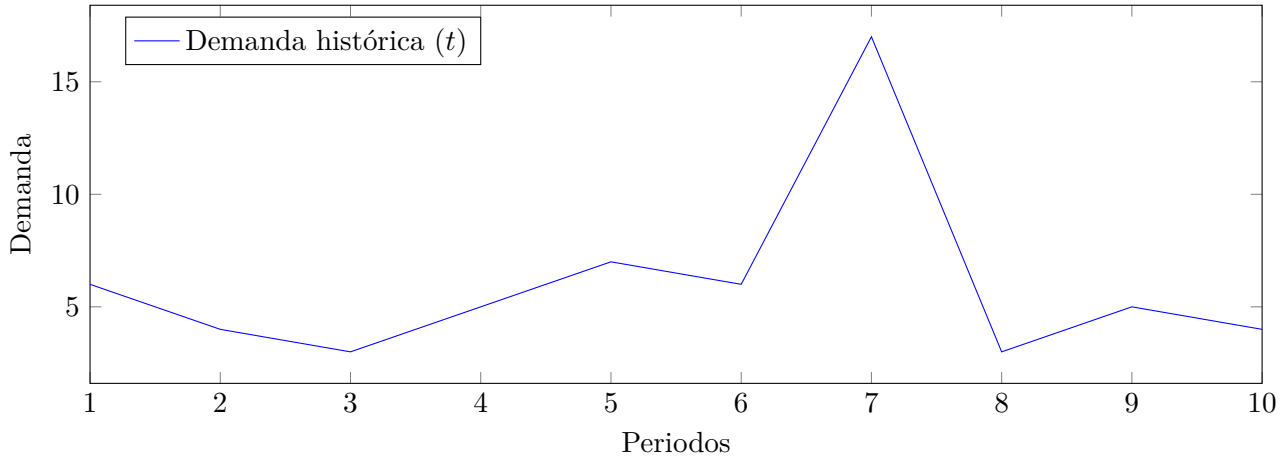


Figura 2.1: Gráfica de la demanda histórica del producto expuesta en la Tabla 2.1.

inferiores y superiores. Estos límites se obtienen de restar y sumar la desviación estándar a la media de las observaciones previas a la observación que queremos analizar.

La desviación atípica o estándar de una variable discreta nos proporciona la variación del conjunto de periodos. Viene dada por la siguiente formulación:

$$s = \sqrt{\frac{\sum_{i=1}^{|t|} (t_i - \bar{t})^2}{|t|}} \quad (2.3)$$

donde:

- t Conjunto histórico de demanda del producto.
- \bar{t} Media aritmética del conjunto.
- $|t|$ Cardinalidad del conjunto t .

Volviendo al ejemplo expuesto en la Tabla 2.1 podemos calcular la desviación típica del subconjunto $m = \{t_1, t_2, \dots, t_6\}$ y determinar si la muestra t_7 es un valor atípico. Al calcular la media del subconjunto m da un resultado aproximado de 4,97 unidades y una desviación típica de 1,97 unidades. Con esto, podríamos determinar que el límite superior es de 6,94 unidades y el límite inferior de 3 unidades. De esta manera concluimos que t_7 es una desviación atípica pues supera el límite superior establecido para el subconjunto m .

Por otra parte, en el periodo t_1 no se podría realizar un pronóstico debido a que no tenemos ningún histórico anterior a él. Esto es una clara desventaja de este tipo de modelo estadístico. El resto de periodos tomarán como predicción *fi* la demanda del periodo anterior t_{i-1} como se ha explicado anteriormente. Ver Tabla 2.2 y Figura 2.2.

Periodo (i)	1	2	3	4	5	6	7	8	9	10
Demanda (t)	6	4	3	5	7	6	17	3	5	4
Predicción (f)		6	4	3	5	7	6	17	3	5

Tabla 2.2: Demanda histórica de la Tabla 2.1 y predicción por pronóstico ingenuo.

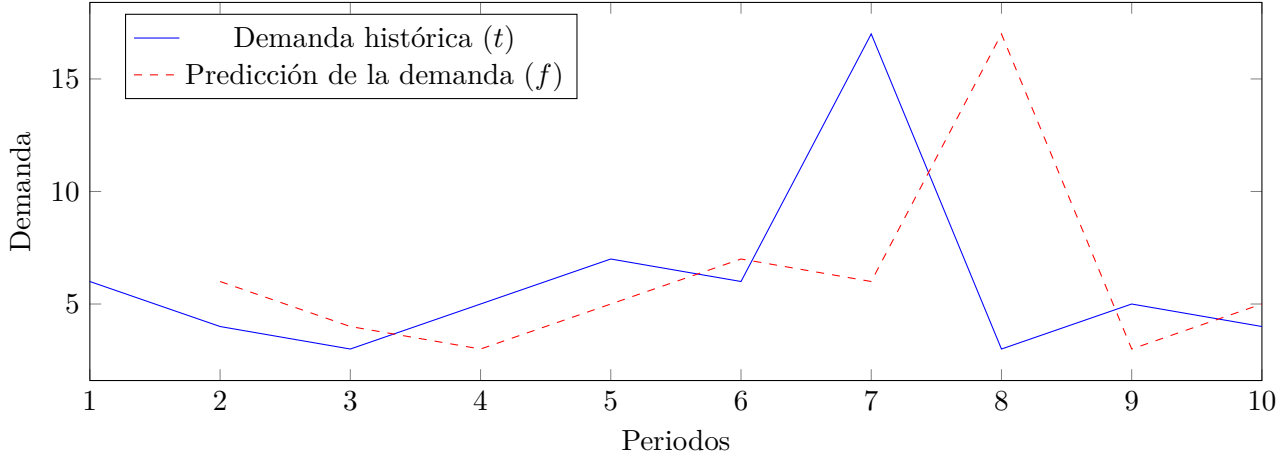


Figura 2.2: Gráfica comparativa de la demanda histórica de la Tabla 2.2 y la predicción por pronóstico ingenuo.

2.2 Media móvil

Este método se basa en una media de los n anteriores periodos para determinar la previsión del siguiente periodo. El modelo de media móvil se basa en la idea de que la demanda futura será similar a las demandas recientes observadas. La predicción de un nuevo periodo f_i dado un conjunto histórico de la demanda t se puede definir de la siguiente manera:

$$f_i = \frac{1}{n} \sum_{j=1}^n (t_{i-j}) \quad (2.4)$$

$$i > n \quad (2.5)$$

donde:

t_i Demanda histórica del periodo i .

f_i Predicción del periodo i .

n Número de observaciones anteriores para calcular f_i .

Entre una de las limitaciones del modelo encontramos la incapacidad de extrapolar cualquier tendencia. Pues en función de los datos de cada empresa la previsión será totalmente distinta.

A continuación, vamos a ver un ejemplo con la demanda histórica de un producto por unidades y un valor n de 3 periodos de demanda. Ver Tabla 2.3 y Figura 2.3.

En este caso nos encontramos con un conjunto de demanda alcista con un cambio a tendencia bajista a partir de t_8 .

Como primera desventaja es que no podemos predecir las observaciones anteriores a $n + 1$,

Periodo (i)	1	2	3	4	5	6	7	8	9	10
Demanda (t)	4	6	8	11	13	14	18	20	15	12

Tabla 2.3: Demanda de un producto en diferentes periodos de tiempo.

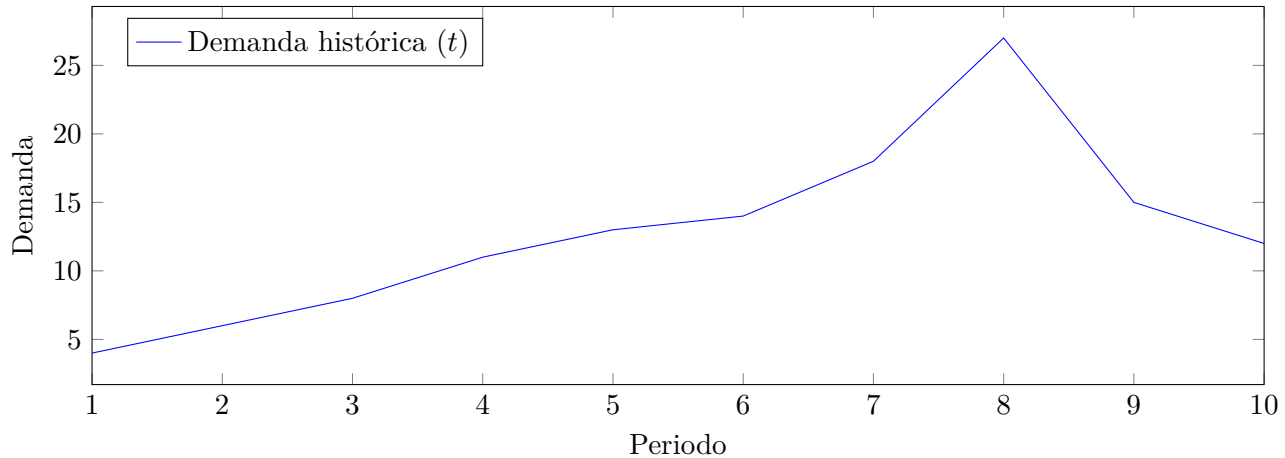


Figura 2.3: Gráfica de la demanda histórica del producto expuesta en la Tabla 2.3.

pues necesitamos la media de n observaciones anteriores para poder realizar el pronóstico.

El resultado de los datos de la Tabla 2.3 con el método de la media móvil se puede ver en la Tabla 2.4 y Figura 2.4.

Periodo (i)	1	2	3	4	5	6	7	8	9	10	11
Demanda (t)	4	6	8	11	13	14	18	27	15	12	
Predicción (f)				6	8,33	10,67	12,67	15	19,33	20	18

Tabla 2.4: Demanda histórica y predicción del producto por el método de la media móvil.

Analizando la Figura 2.2 se puede comprobar que este modelo sigue presentando el problema del ruido cuando existen atipicidades. Al haber ese cambio de tendencia bajista tan brusco produce que los siguientes n valores están sesgados por la muestra t_8 ; no obstante, gracias a la versatilidad del modelo, a partir de t_{11} se adaptaría mejor debido a que t_8 no se tendría en cuenta para la predicción.

2.3 Suavizado exponencial

El suavizado exponencial, explicado en (Vandeput, 2021), corresponde al modelo más sencillo para pronosticar series temporales, y nos permitirá desarrollar modelos muchos más complejos y potentes que los vistos previamente. Su eficacia de basa en que es capaz de recordar el patrón de nivel de demanda. El nivel es el valor medio de fluctuación de la demanda, por lo que se podría considerar como un valor histórico suavizado de la demanda. Es importante entender que no hay una forma matemática de definir el nivel, si no que este se definirá en función de nuestro modelo estimado.

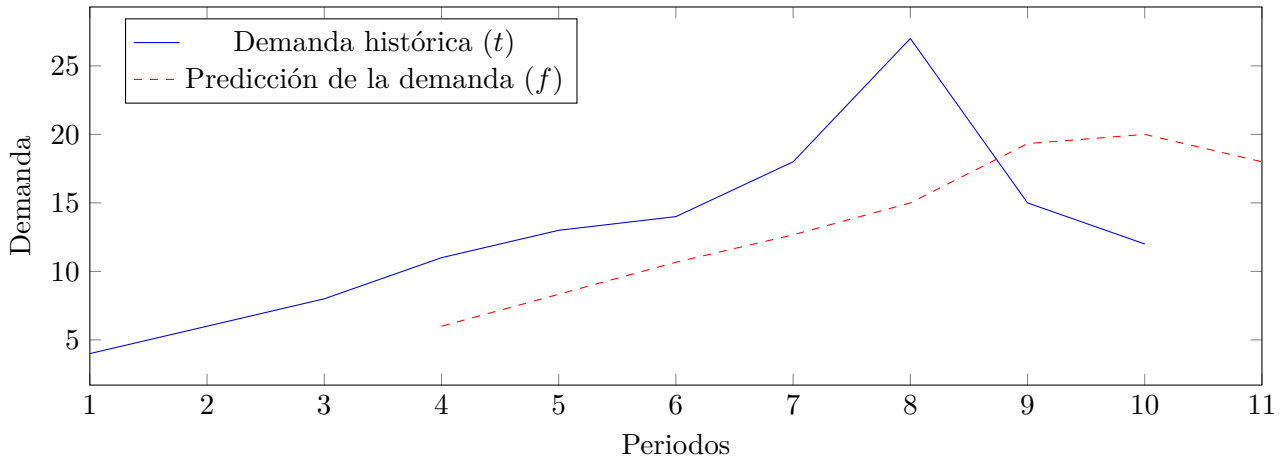


Figura 2.4: Gráfica comparativa de la demanda histórica y la predicción de la demanda resultante del método de la media móvil.

Este modelo presenta varias ventajas con respecto a los métodos anteriores:

- Reduce los valores atípicos, produciendo un menor impacto del ruido que en el pronóstico ingenuo.
- Con respecto a la media móvil, en este modelo se le da más importancia a los periodos más reciente, mientras que los periodos más alejados van perdiendo importancia progresivamente.

El parámetro de suavizado, o tasa de aprendizaje α , determinará cuanta importancia se da a la observación de demanda más reciente. Representemos esto matemáticamente:

$$f_i = \alpha t_{i-1} + (1 - \alpha)f_{i-1} \quad (2.6)$$

$$0 < \alpha \leq 1 \quad (2.7)$$

$$i > 1 \quad (2.8)$$

donde:

- | | |
|-----------------------|---|
| f_i | Predicción de la demanda para el periodo i . |
| t_i | Demanda histórica para el periodo i . |
| α | Porcentaje de importancia en comparación al histórica de la demanda. |
| αt_{i-1} | Representa un valor de importancia sobre la última observación de la demanda. |
| $(1 - \alpha)f_{i-1}$ | Representa cuanto recuerda el modelo de su previsión anterior. |

La bondad de esta fórmula es que la última previsión hecha por el modelo ya está incluyendo una parte de la observación de la demanda anterior y una parte de la previsión anterior. Esto significa que la previsión anterior incluye todo lo que el modelo aprendió hasta ahora basado en el historial de demanda. Ver Figura 2.5.

Si α es alto, el modelo asignará más importancia a la observación de demanda más reciente (aprenderá más rápido). Si el α tiene un valor bajo, el modelo notará menos cambios en el

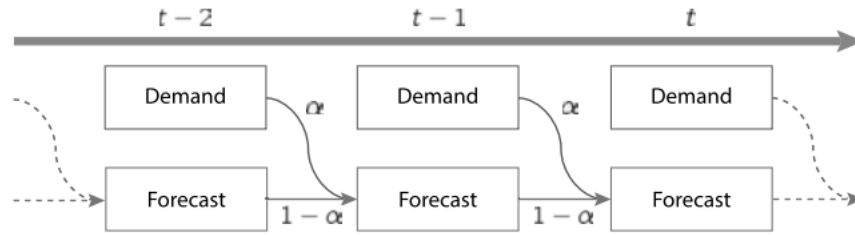


Figura 2.5: Algoritmo de suavizado exponencial simple para la previsión de la demanda (Vandeput, 2021).

nivel, aunque también será menos sensible a las desviaciones y al ruido.

En cuanto a la inicialización del modelo, siempre lo inicializaremos con la primera previsión (período 1) como primera observación de la demanda. Por lo que obtendríamos: $f_1 = t_1$. Esta sería una manera correcta y sencilla de inicializar el pronóstico.

A continuación, veremos un ejemplo con la demanda expuesta en la Tabla 2.5 y representada en la Figura 2.7.

Periodo (i)	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Demanda (t)	5	8	6	8	9	11	12	10	10	13	10	11	12	13	13	13	10

Tabla 2.5: Demanda del producto en diferentes periodos de tiempo.

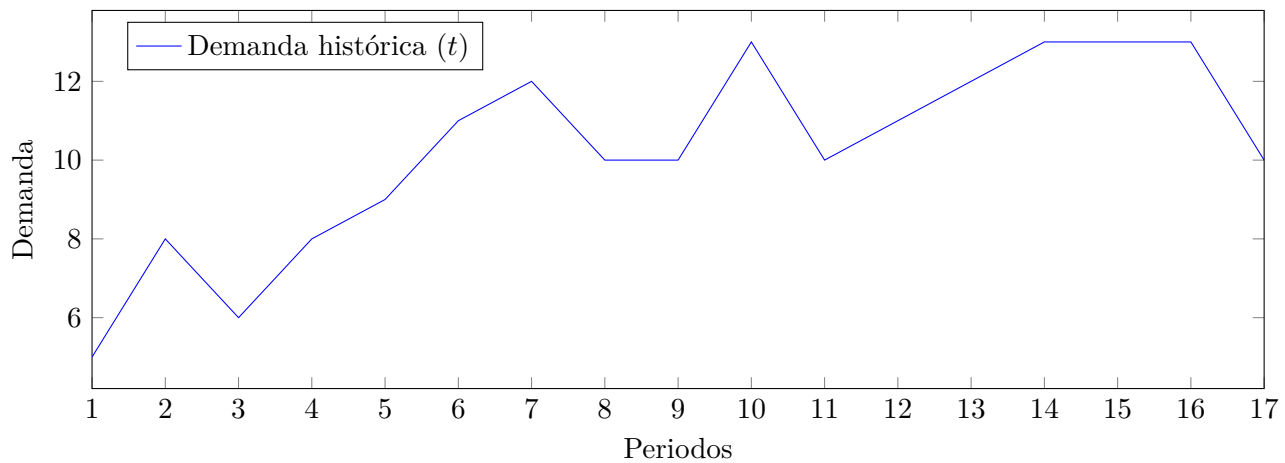


Figura 2.6: Gráfica de la demanda histórica del producto con los datos expuestos en la Tabla 2.5.

Para este ejemplo utilizaremos una tasa de aprendizaje α de 0.2, 0.4 y 0.6 para comparar y observar cómo la tasa de aprendizaje influye en el conjunto de predicción, cuyos resultados se muestran en la Figura 2.7.

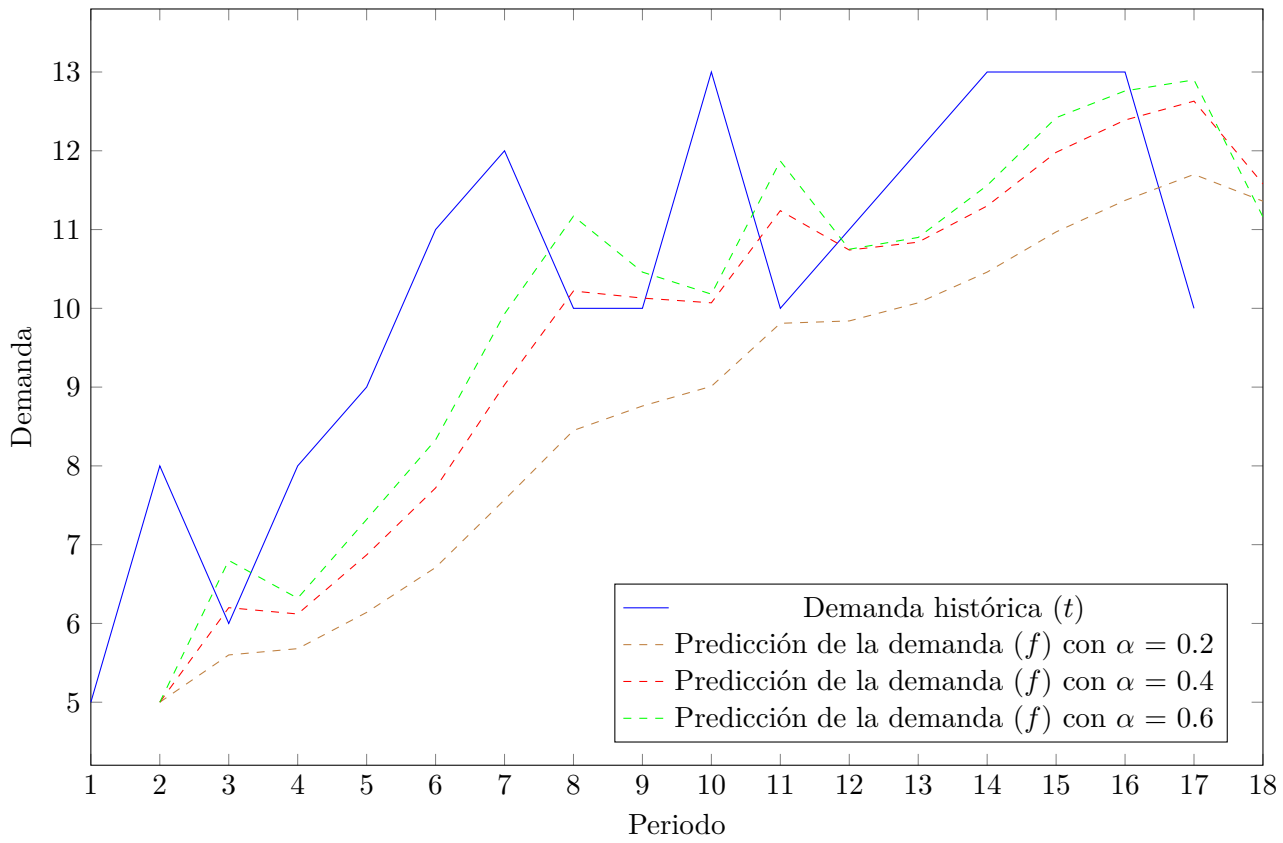


Figura 2.7: Gráfica comparativa de la demanda histórica y la predicción resultante del método de suavizado exponencial simple con los datos de la Tabla 2.5.

Como se puede observar, cuanto mayor es la tasa de aprendizaje menos tiende a suavizarse la tendencia final predicha. Esto es así porque si α es mayor de 0.5 significa que el modelo casi no asigna importancia al histórico de la demanda el pronóstico se basará casi exclusivamente en las últimas observaciones.

En cuanto a las desventajas del modelo, podemos encontrar las siguientes:

- Uno de los principales errores en los que podemos caer a la hora de implementar el modelo es en la sobrecarga de datos. Esta sobrecarga se produce cuando, por ejemplo, inicializamos la previsión como el promedio de los cinco primeros periodos, es decir, recibimos información sobre la demanda futura. Sería como preguntarle a nuestro programa ¿Puede proporcionarme una previsión del próximo período, sabiendo que el promedio de la demanda para los próximos cinco períodos es de 10?. El modelo dará una buena precisión para los períodos iniciales, ya que le hemos dado la demanda promedio de estos períodos, pero no será capaz de replicar tal precisión en el futuro.
- Otro problema, es que el modelo, aunque en menor grado que los métodos anteriores, sigue presentando ruido y desviaciones atípicas ante factores externos.

Es necesario mencionar, que los modelos de suavizado exponencial son un producto del desarrollo de métodos menos complejos. En dichos desarrollos intervinieron algunos ingenieros

como R. G. Brown, autor de lo que sería el primer pronóstico de suavizado en 1956 (Brown, 1956); Charles C. Holt, que refino las ecuaciones de Brown en su artículo (C. Holt, 1957) y finalmente, los modelos culminaron en el suavizado exponencial gracias a las mejoras de Peter Winters (Winters, 1960).

2.4 Suavizado exponencial doble

Este modelo, explicado en (Vandeput, 2021), estimará la tendencia con base en un peso exponencial β , dando más o menos importancia a las observaciones más recientes. Definimos la tendencia como la variación media del nivel de la serie temporal entre dos períodos consecutivos. Si asumimos que una serie de tiempo sigue una tendencia, lo más probable es que no se sepa su magnitud de antemano, por lo que crearemos un modelo que aprenderá por sí mismo la tendencia a lo largo del tiempo.

La idea general de los modelos de suavizado exponencial es que cada vez se añadan más factores que afectan al resultado final. En el caso del suavizado exponencial simple teníamos lo que denominamos como el nivel; en este modelo, tenemos tanto el nivel como la tendencia y; posteriormente, en el modelo de suavizado exponencial triple será el nivel, la tendencia y la estacionalidad.

Para estimar el nivel de demanda lo haremos con la siguiente formulación:

$$a_i = \alpha t_i + (1 - \alpha)(a_{i-1} + b_{i-1}) \quad (2.9)$$

donde:

t	Conjunto de periodos de la demanda histórica.
a	Conjunto de los niveles de la demanda.
b	Conjunto de la tendencia de la demanda.
$a_{i-1} + b_{i-1}$	Representa cuanto aumenta la tendencia desde la previsión anterior.
α	Tasa de aprendizaje del nivel.
β	Tasa de aprendizaje de la tendencia.

Por otra parte, la tendencia se define como:

$$b_i = \beta (a_i - a_{i-1}) + (1 - \beta) b_{i-1} \quad (2.10)$$

donde:

a	Conjunto de los niveles de la demanda
b	Conjunto de la tendencia de la demanda
β	Tasa de aprendizaje de la tendencia

Realmente, estamos aplicando la misma lógica que en el suavizado exponencial simple. El modelo actualizará su estimación del nivel en cada período, gracias a dos datos: la última observación de la demanda histórica t_i y la estimación del nivel anterior aumentada por la tendencia $a_{i-1} + b_{i-1}$.

Una vez definido la estimación del nivel y la tendencia, la previsión de un periodo se fórmula como:

$$f_{i+1} = a_i + b_i \quad (2.11)$$

En este modelo destaca en la relación que existe entre el error de la previsión y la tendencia.

Periodo (i)	1	2	3	4	5	6	7	8	9	10
Demanda (t)	12	20	28	37	44	48	59	66	50	44

Tabla 2.6: Demanda histórica del producto en diferentes periodos de tiempo.

Si la tendencia disminuye cuando el error es positivo y la tendencia aumenta cuando el error es negativo, nuestro modelo está aprendiendo de sus propios errores, algo que los modelos explicados previamente no podían hacer.

A la hora de inicializar el nivel y la tendencia se puede hacer de la siguiente manera:

$$a_1 = t_1 \quad (2.12)$$

$$b_1 = t_2 - t_1 \quad (2.13)$$

A continuación, al igual que hemos hecho en los anteriores métodos, vamos a exponer un ejemplo. Para ello, nos basaremos en la demanda histórica de la Tabla 2.6 y representada en la Figura 2.8.

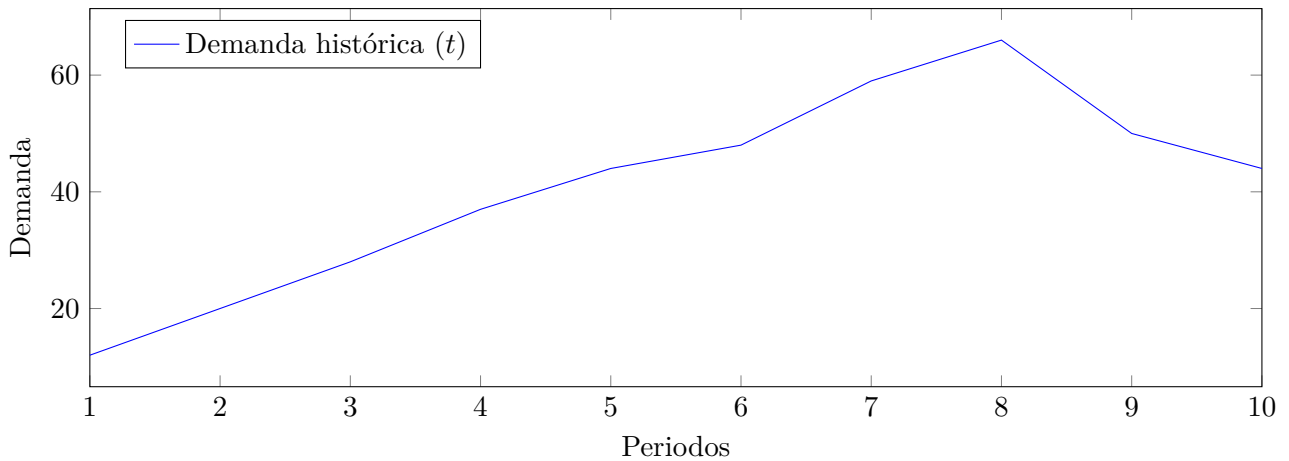


Figura 2.8: Gráfica de la demanda definida en la Tabla 2.6.

Los resultado después de aplicar el algoritmo con un valor de 0.4 para α y β se proporcionan en la Tabla 2.7 y la Figura 2.9.

Periodo (i)	Demanda (t)	Previsión(f)	Nivel(a)	Tendencia(b)	Error($f - t$)
1	12.0		12.000000	8.000000	
2	20.0	20.000000	20.000000	8.000000	0.000000
3	28.0	28.000000	28.000000	8.000000	0.000000
4	37.0	36.000000	36.200000	8.080000	1.000000
5	44.0	44.280000	44.224000	8.057600	-0.280000
6	48.0	52.281600	51.425280	7.715072	-4.281600
7	59.0	59.140352	59.112282	7.703844	-0.140352
8	66.0	66.816125	66.652900	7.638554	-0.816125
9	50.0	74.291454	69.433163	5.695237	-24.291454
10	44.0	75.128401	68.902721	3.204965	-31.128401
11		72.107686	72.107686	3.204965	

Tabla 2.7: Resultados de aplicar modelo de suavizado exponencial doble a la demanda histórica de la Tabla 2.5.

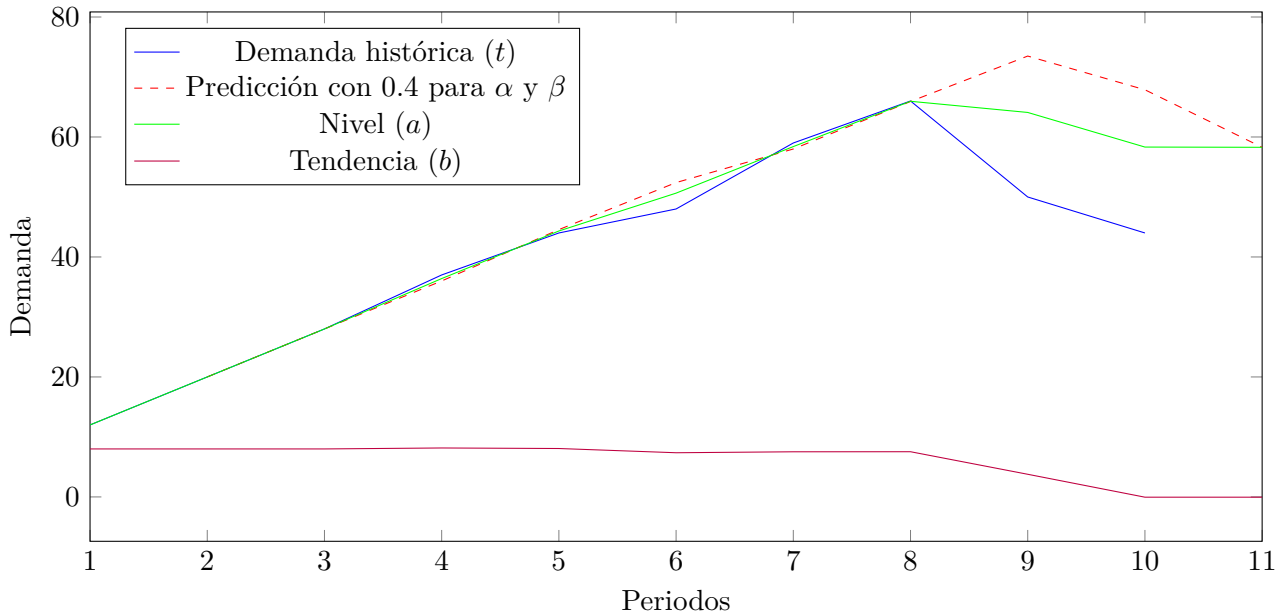


Figura 2.9: Gráfica comparativa de la demanda histórica y la predicción resultante del método de suavizado exponencial doble exponiendo los resultados de la Tabla 2.7.

Como vemos este modelo predice muy bien la demanda excepto para los últimos periodos donde la demanda cambia de tendencia. Un gran inconveniente, es que nuestro modelo considera que la tendencia siempre se mantendrá en el tiempo, lo que no tiene por qué ser así ya que puede haber ciertos factores externos que rompan la tendencia, como, por ejemplo, las estacionalidades. Esto se conoce como el riesgo de la tendencia infinita y fue uno de los motivos por los cual se crearon modelos más avanzados como el del suavizado exponencial triple.

3 Metodología

En el apartado anterior, hemos analizado como han ido evolucionando los modelos estadísticos para predecir la demanda, no obstante, desde hace unos años el auge de los modelos de inteligencia artificial ha ido evolucionando como herramientas con un rendimiento excelente en términos de precisión de pronóstico para un conjunto de datos.

Entre los modelos de aprendizaje automático podemos distinguir dos tipos:

- **Aprendizaje supervisado:** Basan su aprendizaje en conjunto de datos previamente etiquetado. Por un etiquetado se entiende que para cada elemento del conjunto de datos conocemos el valor de su atributo objetivo.
- **Aprendizaje no supervisado:** Basan su aprendizaje en un conjunto de datos sin etiquetas o clases previamente definidas. Es decir, a priori no conocen ningún valor objetivo o de clase, ya sea cuantitativo o cualitativo.

El propósito del aprendizaje automático supervisado es obtener un modelo que pueda realizar predicciones repetibles. Por lo general, no nos importa si el modelo es interpretable o no, es decir, internamente no tenemos que saber como ha realizado las relaciones entre las diferentes variables, solo tenemos que centrarnos en la configuración del modelo y la gestión de los datos. Por ende, el modelado estadístico se trata más de encontrar relaciones entre variables y la importancia de esas relaciones.

Llevando este concepto al campo de estudio de este trabajo, podemos concluir que mientras un modelo estadístico tradicional aplicará una relación predefinida para pronosticar la demanda, un algoritmo de aprendizaje automático no asumirá a priori una relación particular sino que aprenderá estos patrones directamente de la demanda histórica. Esto es de suma importancia pues los modelos como el suavizado exponencial trata cada entrada de manera individual e independiente a las demás mientras que los modelos de aprendizaje automático aprenderán de todo el conjunto de datos.

3.1 Aprendizaje Automático

3.1.1 Árboles de decisión

El aprendizaje del árbol de decisiones (Mitchell y cols., 1997) es un método para aproximar funciones objetivo de valores discretos, en el que la función aprendida está representada por un árbol de decisiones.

Los árboles de decisión clasifican las instancias clasificándolas en el árbol desde la raíz hasta algún nodo hoja, que proporciona la clasificación de la instancia. Cada nodo en el árbol especifica una prueba de algún atributo de la instancia, y cada rama descendente de ese nodo corresponde a uno de los posibles valores para este atributo. La instancia se clasifica comenzando en el nodo raíz del árbol, probando el atributo especificado por este nodo, luego

bajando por la rama del árbol correspondiente al valor del atributo en el ejemplo dado. Este proceso se repite luego para el subárbol arraigado en el nuevo nodo.

Un ejemplo de un árbol de decisión se puede observar en la Figura 3.1. En este ejemplo se estima si una mañana en particular es adecuada para sacar a pasear al perro. Para ello, se utilizan tres variables categóricas: Tiempo, Temperatura y Viento. Los posibles valores de la variable objetivo son Si o No.

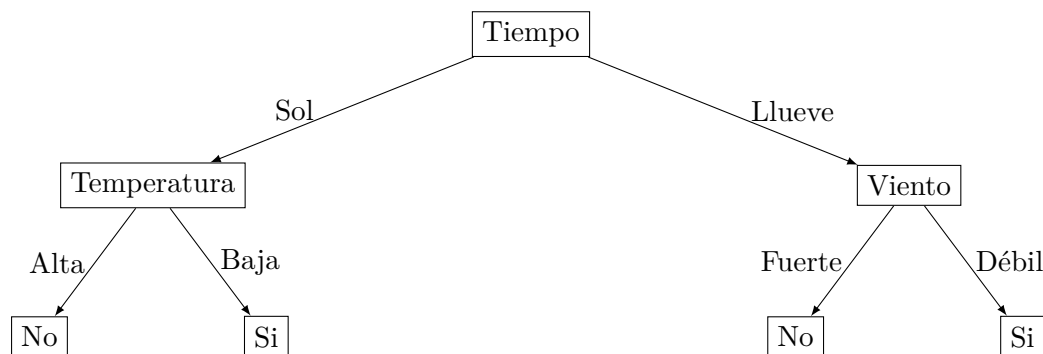


Figura 3.1: Árbol de decisión.

En este punto, distinguimos dos tipos de árboles: árboles de regresión y árboles de clasificación. Los árboles de regresión se utilizarán cuando la variable objetivo es cuantitativa y los árboles de clasificación cuando la variable objetivo es cualitativa:

- Los problemas de regresión requieren una estimación de una salida numérica basada en varias entradas. Por ejemplo, la previsión es un problema de regresión.
- Los problemas de clasificación requieren que clasifique las muestras de datos en diferentes categorías. Por ejemplo, identificar imágenes como gato o perro es un problema de clasificación.

En este trabajo emplearemos únicamente los árboles de regresión. Este tipo de árboles se construye a través de un proceso conocido como partición recursiva binaria (Naylor y cols., 1990), que es un proceso iterativo que divide los datos en particiones o ramas, y luego continúa dividiendo cada partición en grupos más pequeños a medida que el método avanza en cada rama. Para ello, hará preguntas de tipo $ijk < c$? para cada una de las covariables, de esta forma el espacio de las covariables es dividido en una partición y todas las observaciones que queden dentro de una partición tendrán el mismo valor estimado.

En la Figura 3.2 se ilustra un árbol en el lado izquierdo y la partición del espacio en el lado derecho. Como se ha explicado previamente, la partición del espacio se hace de manera repetitiva para encontrar las variables y los valores de corte c de tal manera que se minimice la función de coste.

Por otro lado, es importante definir un criterio para el crecimiento del árbol. Si no se establece un criterio el árbol intentará crecer hasta llegar a minimizar la función de coste a cero y se producirá el fenómeno conocido como sobreajuste. Este fenómeno pasa cuando el modelo se ajusta demasiado a los datos de entrenamiento y tiene un excelente rendimiento en estos, pero cuando encuentre con datos que no nunca ha visto dará una precisión muy deficiente.

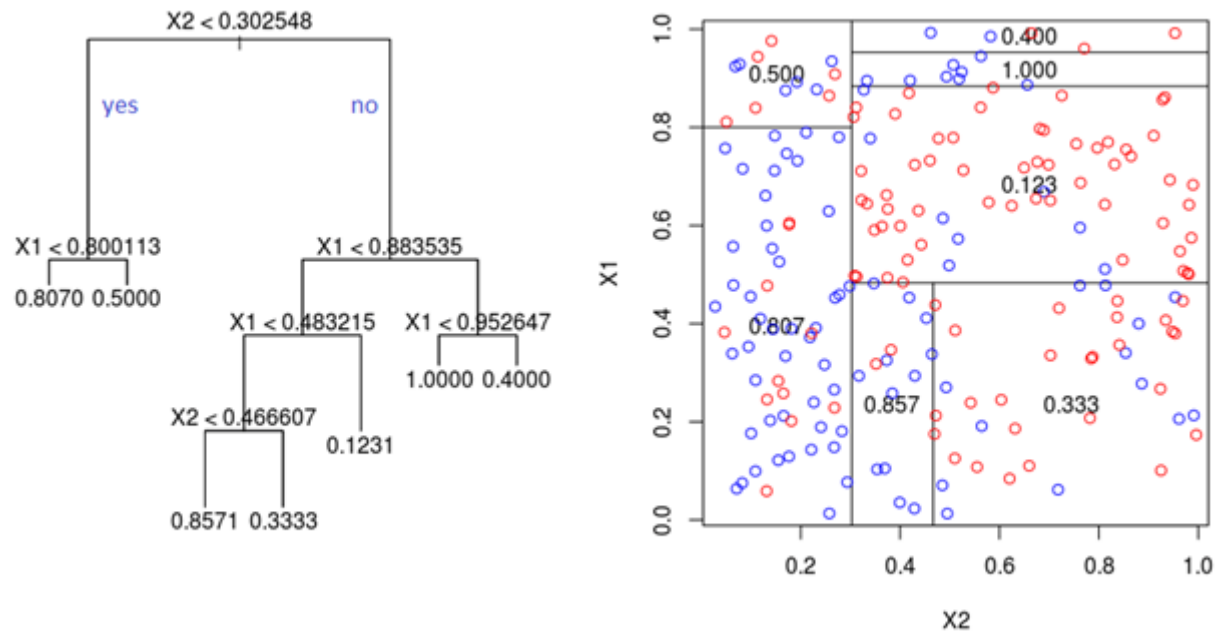


Figura 3.2: Árbol de regresión y sus particiones en un hiper rectángulo.

Para evitar este crecimiento ilimitado, se deben tener en cuenta una serie de parámetros que permitirá ajustar el modelo para crecer solo lo necesario para no llegar al sobreajuste. Entre los parámetros más importantes destacan:

- **Máxima profundidad** : La cantidad máxima de preguntas consecutivas (nodos) que puede hacer el árbol. Más preguntas significa que el algoritmo puede hacer un mejor trabajo para distinguir los casos en el conjunto de entrenamiento y, en adelante, lograr una mayor precisión, pero demasiados nodos desembocará en un sobreajuste.
- **Mínima cantidad de muestras para dividir** : La cantidad mínima de muestras que se requieren en un nodo para activar una nueva división. Reducir esta cantidad hará que se logre un mayor número de particiones y por tanto mayor precisión, no obstante, al igual que con el parámetro de máxima profundidad, conseguir demasiadas particiones hará que surja sobreajuste.
- **Mínima cantidad de observaciones por hoja** : La cantidad mínima de observaciones que necesita estar en una hoja. Cuanto más cerca esté de 0, mayor será el riesgo de sobreajuste: el árbol crecerá hasta que haga suficientes preguntas para tratar cada observación por separado.

3.1.2 Bosques aleatorios y extremadamente aleatorios

Los bosques aleatorios, o los bosques de decisiones aleatorias, es un método de aprendizaje automático para los problemas de clasificación y regresión. Consiste en un grupo de árboles de decisión individuales que operan como un conjunto.

Cada árbol individual en el bosque aleatorio tiene como salida una predicción de la clase o del valor y la predicción con más votos se convierte en la predicción de nuestro modelo. Ver Figura 3.3.

El concepto fundamental detrás del bosque aleatorio es simple: la sabiduría de las multitudes (Surowiecki, 2005). Esta idea explica que la opinión promedio de un grupo de personas va a ser más precisa que la opinión de un solo miembro del grupo. En las empresas de distribución de alimentación y bebidas un ejemplo simple podría ser el siguiente: si se desea hacer una compra a un proveedor para el próximo mes y deseamos saber cuántas unidades de cada producto se deben comprar, es mejor pedir la opinión de muchos miembros diferentes del equipo (comerciales, administración, gerente, analistas financieros, etc) y tomar el promedio de los diferentes pronósticos en lugar de confiar ciegamente en un solo miembro del equipo.

En términos de ciencia de datos (Yiu, 2019), la razón por la que el modelo de bosque aleatorio funciona es debido a que un conjunto de árboles relativamente no correlacionados darán mejores resultados que cualquiera de los individuos del conjunto individualmente, siguiendo la hipótesis de la sabiduría de las multitudes. La baja correlación entre los árboles de decisión es el punto clave. La razón de efecto es que los árboles se protegen unos a otros de sus errores individuales siempre que no se equivoquen constantemente en la misma dirección. Aunque algunos árboles puedan estar equivocados, otros árboles serán correctos, por lo que, como grupo, los árboles pueden moverse en la dirección correcta.

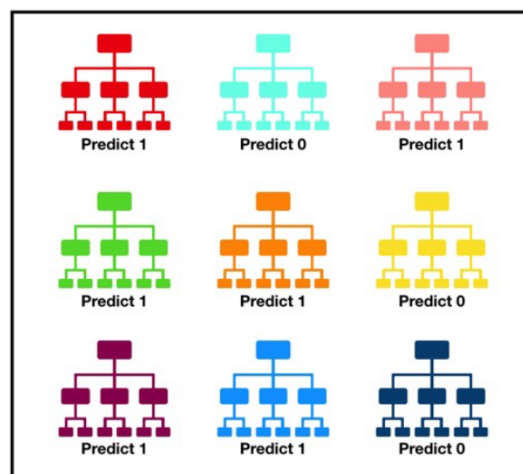


Figura 3.3: Funcionamiento de un bosque aleatorio (Yiu, 2019).

Además de los bosques aleatorios existen los bosques extremadamente aleatorios (Geurts y cols., 2006) que llevan la aleatoriedad un paso más allá. A la hora de escoger una característica y un valor de corte c para dividir cada nodo, en lugar de escoger el valor de corte c que mejor divida cada característica y la característica y valor de corte que minimice el criterio de impureza, este modelo genera un valor de corte aleatorio para cada característica planteada, escogiéndose como regla de división el mejor de ellos. Por tanto, la diferencia clave entre un bosque aleatorio y uno extremadamente aleatorio es que el primero elige la división mejor mientras que el segundo la elige al azar. Esto hace que el coste computacional sea menor al ahorrar el cálculo de la división más óptima y que se reduzca el sesgo como la varianza.

3.1.3 AdaBoost

AdaBoost es el acrónimo para el método "Adaptive Boosting" (Freund y Schapire, 1997) y es un modelo de conjunto de impulso. Este algoritmo busca crear un clasificador fuerte cuya base sea la combinación secuencial de clasificadores débiles. Para ello, entrena una serie de clasificadores débiles de manera iterativa de modo que cada nuevo clasificador utilice como retroalimentación los datos que fueron erróneamente clasificados por su predecesor. De esta manera, el algoritmo se adapta y logra tener mejores resultados. Entendemos por clasificador débil, un clasificador binario; es decir, nos dicta si una variable x pertenece a una clase o no. Por otra parte, un clasificador fuerte es multiclase: utiliza un conjunto de clasificadores débiles para determinar a qué clase pertenece x del conjunto de clases posibles.

Como hemos mencionado, este modelo utiliza una técnica de conjunto de impulso. Esta, es una técnica de conjunto en la que se entrenan varios modelos individuales de forma secuencial. En cambio, el modelo previamente explicado, bosques aleatorios, utiliza una técnica de embolsado. Ésta última se basa en el entrenamiento de varios modelos individuales de forma paralela.

Un ejemplo sencillo del funcionamiento de Adaboost se da en (Freund y cols., 1999): considera un filtro capaz de diferenciar entre un correo spam y uno que no lo es: es decir, la clase sería: Spam / No Spam. Para esto, se crean clasificadores débiles acerca de qué tipo de palabras clave contienen este tipo de archivos y así generar un clasificador que logré diferenciar de manera más precisa entre estos dos tipos de correo.

Como hemos visto, Adaboost es un algoritmo de clasificación, pero, como hemos mencionado anteriormente, estamos ante un problema de regresión. Entonces, ¿cómo puede Adaboost ayudarnos en nuestra labor de predecir el stock de un producto en un periodo?. Para ello se juntará el modelo Adaboost y los árboles de decisión; en este contexto, el algoritmo intentará hacer crecer secuencialmente los árboles de decisión como clasificadores débiles y castigar las predicciones predichas incorrectamente asignándoles un peso mayor después de cada ronda de predicción. Así, la decisión final será el voto de la mayoría de ponderada por los pesos asignados a cada clasificador débil. Ver Figura 3.4.

En el algoritmo Adaboost tenemos diferentes parámetros a destacar que deberemos que tener en cuenta a la hora de configurarlo:

- **Clasificador base:** En este trabajo utilizaremos como un clasificador base un árbol de regresión que nos permitirá predecir valores numéricos. Cabe destacar, que este árbol de regresión se deberá de ajustar previamente teniendo los parámetros mencionados en el apartado 3.1.1 para evitar un posible sobreajuste.
- **Tasa de aprendizaje:** Un coeficiente que reduce el impacto general de cada clasificador débil en la ponderación del conjunto de entrenamiento.

3.1.4 Gradient Boosting y Extreme Gradient Boosting

En 2001, Jerome H. propuso un nuevo concepto para impulsar árboles: Gradient Boosting (Friedman, 2001). La finalidad es parecida al algoritmo Adaboost sólo que la diferencia principal erradica en que en Adaboost minimiza exclusivamente una única función de pérdida: la función de pérdida exponencial, y puede hacer que el algoritmo sea sensible a los valores atípicos. Con Gradient Boosting, se puede utilizar cualquier función de pérdida diferenciable.

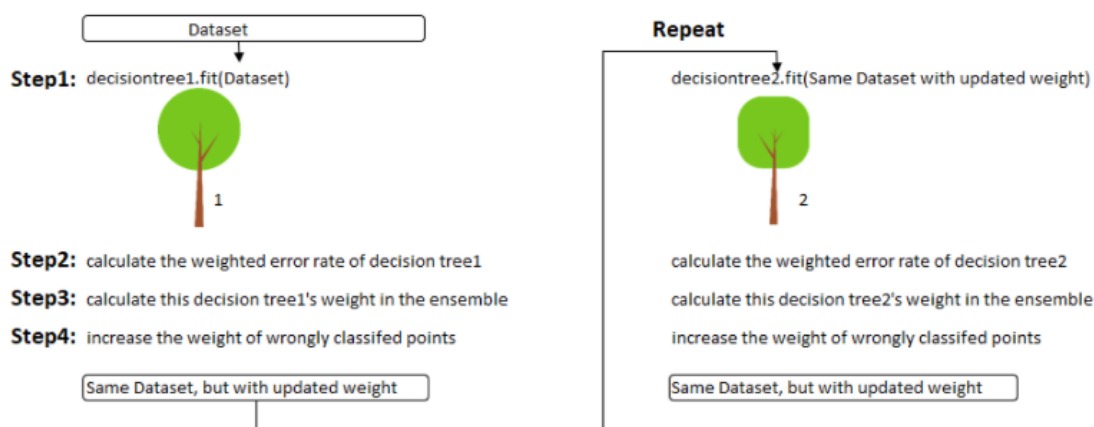


Figura 3.4: Funcionamiento del algoritmo Adaboost con árboles de decisión (Cheng, 2019).

El algoritmo Gradient Boosting es más robusto para valores atípicos que AdaBoost y, por ello, es una buena opción como estudio para el problema expuesto.

En este trabajo me centraré en el uso de exclusivo de Extreme Gradient Boosting. Tanto Extreme Gradient Boosting como Gradient Boosting siguen el principio de aumento de gradiente. Sin embargo, existe la diferencia en los detalles del modelado. Específicamente, Extreme Gradient Boosting utiliza una formalización de modelo más regularizada para controlar el sobreajuste, lo que le da un mejor rendimiento. El nombre Extreme Gradient Boosting, sin embargo, en realidad se refiere al objetivo de ingeniería de empujar el límite de recursos de cálculo para algoritmos de árbol potenciados.

3.2 Aprendizaje Profundo

El aprendizaje profundo se fundamenta principalmente en un modelo computacional denominado redes neuronales.

Uno de los modelos que más clásicos del aprendizaje profundo es el perceptrón multicapa y será el que estudiamos para su posterior aplicación.

El perceptrón (Rosenblatt, 1958) es la unidad más básica de un perceptrón multicapa y su función es la de modelar el comportamiento de una neurona "natural". Para ello, recibe un conjunto de parámetros como entrada a los que asocia un peso, realiza una suma ponderada de éstos teniendo en cuenta el peso de cada uno y finalmente, aplica una función no lineal denominada función de activación al resultado de esa suma ponderada obteniendo una salida. Ver Figura 3.5.

El perceptrón multicapa es la unión de capas verticales formadas por perceptrones. Cada capa está unida con otras a través de enlaces. En estos enlaces el valor de salida del perceptrón de la capa anterior es multiplicado por un valor de peso. Estos pesos en los enlaces puede incrementar o decrementar el estado de activación de las neuronas adyacentes. Además, este modelo utiliza una técnica llamada retropropagación para el entrenamiento (Sathyanarayana,

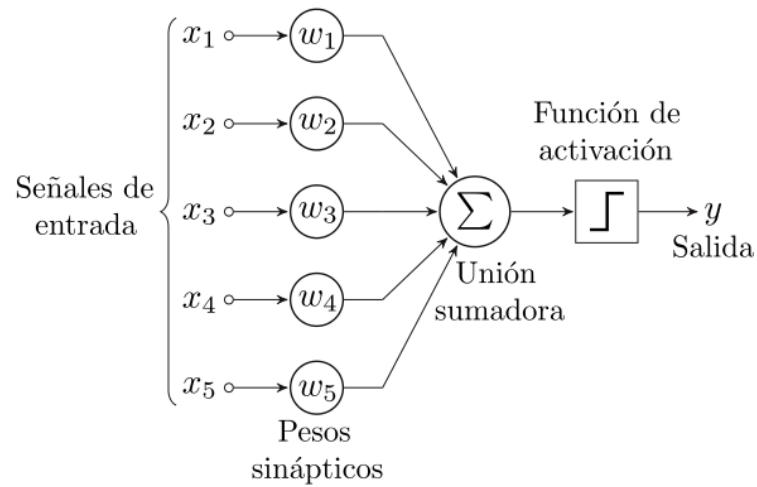


Figura 3.5: Diagrama de un perceptrón con cinco señales de entrada (Cartas, s.f.).

2014). Ver Figura 3.6.

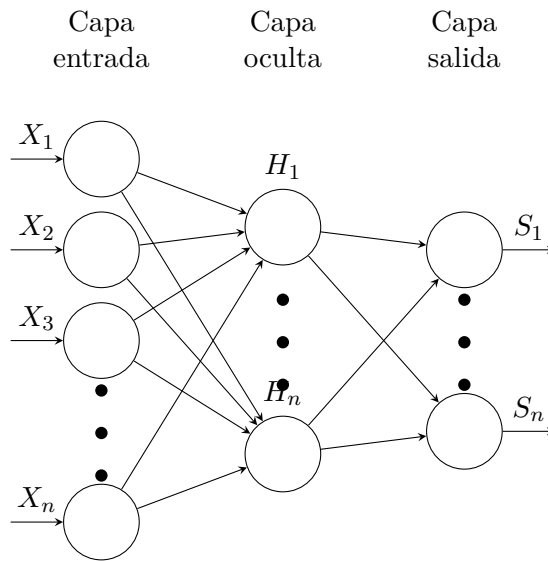


Figura 3.6: Funcionamiento de un perceptrón multicapa

Las capas pueden clasificarse en tres diferentes tipos:

- **Capa de entrada:** Esta capa está compuesta por aquellos perceptrones que introducen los patrones de entrada en la red.
- **Capas ocultas:** Formada por perceptrones que tienen como entrada el resultado de capas anteriores y dan como salida un resultado a las capas posteriores.
- **Capa de salida:** La salida de los perceptrones de esta capa darán el resultado del

modelo.

El funcionamiento de un perceptrón multicapa es el siguiente: los datos de entrada van de una capa a otra a través de los diferentes perceptrones en un proceso denominado propagación hacia adelante. Esta es la fase en la que los perceptrones propagan información a través de la red aplicando sus funciones de activación a sus entradas ponderadas y sesgos. Durante esta fase, la red neuronal generará sus predicciones para posteriormente utilizar el proceso de retropropagación que se utilizará para ajustar los pesos en función de los errores de la predicción generada por la propagación hacia adelante. En la fase de entrenamiento, este proceso se va haciendo por turnos consiguiendo ajustar los pesos de cada perceptrón y conseguir generalizar lo suficiente el modelo para poder predecir correctamente.

4 Desarrollo

En los capítulos anteriores hemos introducido el estado de arte en la predicción de stock, así como las metodologías estadísticas usualmente aplicadas en las empresas de la distribución de alimentos y bebidas. Asimismo, hemos presentado diferentes metodologías basadas en inteligencia artificial para la predicción de stock y su potencial para ser empleadas en empresas de este sector. En este capítulo mostraré el desarrollo de estos modelos y su aplicación concreto a partir de los datos suministrados por dos empresas de la distribución de alimentos y bebidas. Presentaré, además, la gestión de datos, así como los hiperparámetros y ajustes en los modelos utilizados. Antes, evaluaremos los diversos factores internos y externos que condicionan la predicción de stock, lo que nos permitirá percibir mejor la complejidad de la predicción de stock a las que se enfrentan las empresas de este sector. Como veremos, son muchos los parámetros a tener en cuenta; en particular, los parámetros externos son diversos, complejos y difíciles de abordar con rigor en el marco de un Trabajo de Fin de Grado. Por ello, en este trabajo sólo consideraré los parámetros internos para el entrenamiento de los modelos de inteligencia artificial seleccionados.

4.1 Predicción de stock

Antes de explicar cómo se han aplicado los modelos seleccionados, es importante entender por qué no es sencillo realizar una predicción de stock. En esta sección se analizarán los factores determinantes que hacen de la predicción de stock un problema complejo, lo que nos permitirá seleccionar las variables más relevantes para el tratamiento de los datos y el entrenamiento de los modelos.

En los capítulos anteriores hemos visto que para predecir el periodo de demanda futuro los modelos estadísticos se basaban exclusivamente en la demanda histórica. Sin embargo, en el sector de las distribuidoras de alimentación y bebidas predecir la demanda futura no es suficiente. Como explicamos en el Capítulo 1, entendemos por periodo de demanda el stock mínimo de un producto que hay tener en el inventario para poder abastecer la demanda en un determinado periodo de tiempo. Además, para poder estimar la previsión de stock final necesaria, hay que tener en cuenta otros conceptos. Entre ellos:

- **Stock mínimo** (s_{min}): Cantidad mínima del producto que se quiere tener en el almacén. Este concepto se denomina también stock de seguridad y se utiliza para asegurar el stock del producto ante posibles variaciones de la demanda.
- **Stock máximo** (s_{max}): Cantidad máxima de producto que se quiere tener en el almacén.
- **Stock pendiente de recepción** (s_r): Es el stock de un producto que se ha pedido a un proveedor en un periodo anterior pero que aún no se ha recepcionado y, por tanto, no está disponible en el almacén.

De esta manera, la previsión de stock final de stock la podemos definir de la siguiente manera:

$$d_i = \max(\min(f_i - s_r, s_{\max}), s_{\min}) \quad (4.1)$$

$$s_{\max} > s_{\min} \quad (4.2)$$

d_i	Previsión de stock de un periodo i .
f_i	Predicción de la demanda para un periodo i .
s_{\min}	Stock mínimo que se quiere del producto.
s_{\max}	Stock máximo que se quiere del producto.
s_r	Stock pendiente de recepción.

En este trabajo entrenaré los modelos con el objetivo de que, dado una demanda histórica de un producto y considerando una serie de filtros dependientes de ciertos factores, los modelos pronostiquen el periodo de demanda futuro, que es la variable f_i de la fórmula anterior.

4.1.1 Factores

Como se ha argumentado anteriormente, en las empresas de distribución y bebidas existen factores que van más allá de la demanda histórica, pero, ¿qué otros factores son los que se deberían considerar para predecir la demanda futura?. Para ello, distinguiremos dos tipos:

- **Internos:** Son los factores que controla directamente la empresa. Algunos ejemplos serían: promociones de productos, cambios de precios, etc.
- **Externos:** Son aquellos factores que la empresa no controla. Este tipo de factores se denominan también indicadores adelantados de la economía (Burns y Mitchell, 1946) en el sector de la alimentación y bebidas. Algunos de estos factores serían: el crecimiento del PIB, el crecimiento del IPC en el sector, la tasa de desempleo en el sector, etc.

4.1.1.1 Factores Internos

4.1.1.1.1 Promociones de productos

Uno de los factores que controlan de manera directa las empresas son las promociones. Las promociones en las empresas del sector de la alimentación y bebidas pueden ser de diferente naturaleza: por producto, por familia de productos, por acumulación de puntos, por clientes, etc. El objetivo de las promociones durante un periodo determinado es el de impulsar las ventas de un producto o un conjunto de productos, y fidelizar o captar nuevos clientes. Por tanto, este factor puede influir a la hora de predecir el stock. Por ejemplo, imaginemos que para predecir el stock de un producto tenemos en cuenta la demanda histórica de las últimas cinco semanas. Si en los dos últimos periodos ha existido una promoción del producto lo más probable es que sus ventas se hayan incrementado en esos periodos generando un sesgo a la hora del entrenamiento del modelo.

4.1.1.1.2 Rapples sobre ventas

Este factor se aplica más a las empresas del sector especializadas en bebidas. Un *rappe* es un tipo de descuento que una distribuidora ofrece a sus clientes cuando estos llegan a un determinado volumen de compras, generalmente anuales. Un ejemplo de un *rappe* de una distribuidora a un cliente se muestra en la Tabla 4.1. En esta tabla se puede observar que dependiendo del volumen de compras anual que haga, se le aplicará un descuento en sus compras de cara al año siguiente.

Volumen	Descuento (%)
1000	2
2000	3
5000	5
10000	8

Figura 4.1: Ejemplo de un *rappe* con sus correspondientes tramos

Los *rappe*s se pueden aplicar a productos individuales o a una familia de productos. Este tipo de descuentos muchas veces sólo se aplican a determinados clientes y su importancia en la previsión de stock puede radicar en los tramos. Vamos a suponer que un cliente tiene un *rappe* para un producto determinado con una distribuidora como el de la Tabla 4.1. A principios de cada año se hace el recuento para ver el volumen de compras que ha realizado el cliente y comprobar que descuento se le debe aplicar para futuras compras. Imaginemos ahora que el cliente ha comprado un total de 9500 unidades a lo largo del año, pero no llega al siguiente el tramo y están en el último periodo antes del recuento anual. Una estrategia para este cliente es comprar 500 unidades para que durante el año siguiente se aplique un descuento del 8% lo que a la larga le saldrá más rentable. Por tanto, el *rappe* puede generar atipicidades en los periodos de demanda y es un factor que hay que tener presente a la hora de tratar los datos.

4.1.1.1.3 Precio

En este tipo de empresas es muy común que se utilicen métodos de valoración de existencias. La valoración de existencias se define como: "Conjunto de acciones para la asignación del valor más correcto a las existencias de una empresa, una vez comprobada la existencia física mediante el inventario". Estos métodos son necesarios debido a que los precios de compra varía para el mismo proveedor; es decir, puede que en el almacén exista un stock de un determinado producto que se haya comprado a diferentes precios al proveedor. Imagina que este stock se obtuvo a precios diferentes durante dos compras al mismo proveedor, entonces, ¿a qué precio se debería vender el producto para que no se haga a un margen de venta inferior para la distribuidora?. Para afrontar este problema surgen los métodos de valoración de existencias como el precio medio ponderado, FIFO y LIFO, que permiten calcular un precio teniendo en cuenta los precios de entrada del producto.

El precio afecta directamente a la previsión del stock, puesto que si una empresa realiza el recalcu de los precios con algún método de valoración de existencias y en algún producto hay una variación importante en el precio, esto puede influir posteriormente en la decisión de compra de un cliente en ese producto tanto si hay disminuido o incrementado su precio.

4.1.1.1.4 Depósitos

Por último, hablaremos de otro factor interno importante en las empresas de este sector: los depósitos. Los depósitos son la adquisición por adelantado de existencia de un producto por un cliente para que se le sirva de manera recurrente. Es decir, facturar un producto por adelantado al cliente. Este factor altera las ventas y, por tanto, provoca oscilaciones en los periodos de demanda. Un ejemplo sería el siguiente: un cliente adquiere 100 unidades por depósito de un producto a la distribuidora en un periodo x de tiempo y esta se las factura; por tanto, en su contabilidad ésta ha vendido 100 unidades. Ahora bien, todas esas unidades no se suministran de una vez al cliente, sino que se suministran en función de la demanda del cliente hasta completar el depósito. De esta manera, en el periodo x de tiempo existirá una demanda sesgada por este factor no representando una demanda real.

4.1.1.2 Factores Externos

4.1.1.2.1 Familia del producto y estacionalidades

El tipo de producto es uno de los factores más importantes a la hora de predecir el stock. Como ya hemos mencionado, lo que realmente calculamos es una predicción de demanda de un periodo futuro para un producto concreto; en este punto es donde la familia del producto es un factor externo a considerar. Las familias de productos dependen generalmente de la época del año. Por ejemplo, la familia de los helados tienden a tener una demanda mucho más fuerte en verano que el resto del año. Por ello, no sólo hay que considerar los periodos anteriores de demanda del mismo año, sino que hay que tener en cuenta cómo los periodos posteriores a ese mismo periodo del año anterior afectaron a la demanda. De esta manera, se podrá conseguir reducir las desviaciones atípicas del producto pues se basará en la tendencia de años anteriores. Como se puede ver en el ejemplo de la Figura 4.2 para un determinado producto existen los mismos patrones de tendencia de un año a otro.

Esta es una posible solución para combatir las estacionalidades de un producto, pero ¿será necesario que los modelos sepan además de qué familia es el producto?. No realmente, ya que además surgiría otro problema pues la relación entre el tipo de producto y su consumo tiene que ver también con la zona geográfica. Un ejemplo sería la fiesta de la tomatina. Este evento pasa en una zona geográfica determinada y hace que la media del consumo de la familia de las verduras crezca de manera considerable, pero sólo en la zona geográfica en la que se celebra. Esto es de gran importancia pues si hacemos que el modelo tenga en cuenta la familia del producto del cual está prediciendo el stock podría haber aprendido que en los periodos del año cercanos a esa fiesta, la predicción del stock de la familia de las verduras deberían tener una tendencia alcista; no obstante, esto es una información totalmente sesgada, ya que, sólo pasa en una determinada zona geográfica.

4.1.1.2.2 Clima

Otro factor externo que influye indirectamente es el clima. Este factor está estrechamente relacionado tanto con el tipo de producto y la zona geográfica. La temperatura que ha hecho de media en un periodo de tiempo podría llegar a influir en las ventas. Un ejemplo claro

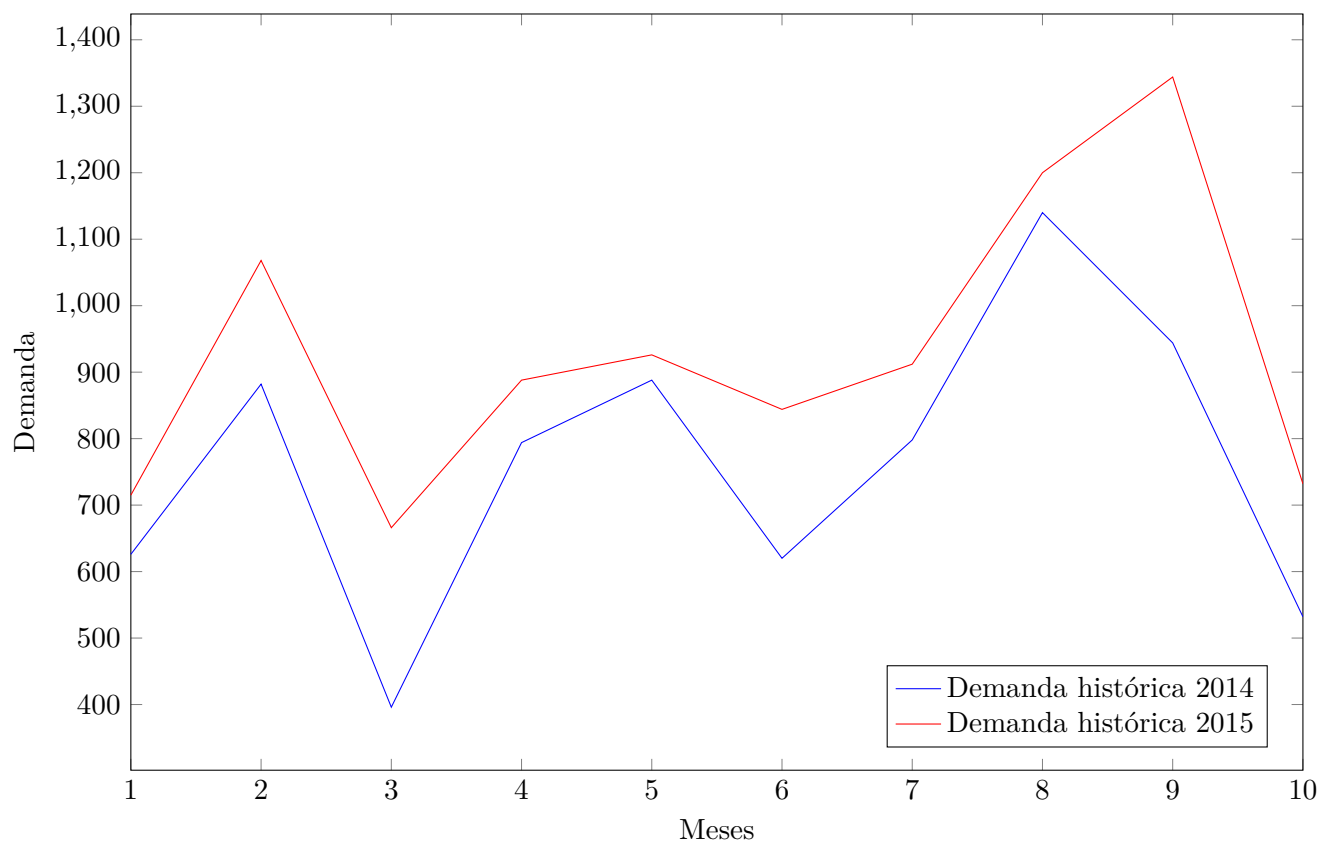


Figura 4.2: Gráfica comparativa de la demanda histórica de un producto en diferentes años.

sería el siguiente: imaginemos una distribuidora de alimentación que su producto principal es el pescado y sus clientes son en gran mayoría restaurantes al pie de playa de una zona geográfica concreta. Esta empresa incrementa la demanda de pescado en verano debido a la estacionalidad de compra de sus clientes, ahora bien, si en uno de esos periodos hay un temporal de lluvia y vientos fuertes venderá menos, ya que sus clientes principalmente venden más cuando hace un buen clima.

4.1.1.2.3 IPC

IPC (Índice de precios al consumo) o inflación es un indicador que mide la variación de los precios de la cesta de bienes y servicios en un lugar concreto durante un determinado periodo de tiempo. Este indicador a priori no parece tener relación aparente con el problema, pero nada más lejos de la realidad. El INE cada año publica la variación del IPC mensual con respecto al mes anterior de los alimentos y bebidas no alcohólicas. Esta variación entre periodos podría ser un indicador válido para el entrenamiento de nuestros modelos, ya que la inflación es un factor macroeconómico que afecta de manera directa al consumidor final (Sarker y cols., 2000). Una empresa de distribución de alimentación y bebidas al final es una empresa intermediaria en el proceso de adquisición de un bien o servicio a un consumidor final, es decir, están a la merced de ellos. Si el consumo del consumidor disminuye en alimentación

y bebidas, las ventas de las distribuidoras de este sector disminuirán; en este contexto, la inflación es un punto clave en la economía del país que junto a otros factores determinan de una manera u otra la cantidad de consumo de una persona, y por ende, se relaciona de una forma indirecta con la previsión de stock. Posiblemente, este factor externo afecte más a las ventas de toda la distribuidora que las de un producto concreto; sin embargo, al existir una relación es un posible factor a tener en cuenta aunque posiblemente no con tanta importancia como las de otros factores previamente explicados.

Como hemos visto todos los factores están íntegramente relacionados y afectan de una manera u otra a la hora de predecir el stock; no obstante, los factores externos van un paso más allá y son más complicados de manejar al no ser controlados directamente por la empresa. Por ello, y, para simplificar el estudio, no se tendrán en cuenta de cara al entrenamiento de los diferentes modelos aunque el análisis realizado nos hace entender **la dificultad del problema expuesto**.

4.2 Gestión de los datos

4.2.1 Obtención de los datos

Los datos que se han utilizado para este trabajo son dos bases de datos de dos empresas de distribución de alimentación y bebidas. Estos datos se consiguieron gracias a una colaboración por parte de la empresa que proporciona el software ERP a estas empresas y que me puso en contacto con ellas.

Ambas utilizan un software ERP especializado en el sector el cual utiliza un motor SQL-Server para almacenar los datos. El primer paso fue estudiar el diagrama de la base de datos para saber dónde se almacenaban los diferentes valores que se necesitaban para este estudio. Principalmente se utilizaron las tablas que almacenaban las fichas de los productos para extraer sus características y las líneas de venta de los albaranes para poder obtener un histórico de la demanda.

El segundo paso fue realizar el desarrollo de un programa que se conectará a las diferentes bases de datos a partir de un archivo de configuración para poder extraer los datos. Una vez extraídos, el propio programa se encargó de estructurar los datos y procesarlos.

4.2.2 Procesamiento y estructura

En este apartado veremos cómo se han procesado los datos y la estructura final del conjunto de datos para el entrenamiento de los modelos. El objetivo es predecir la demanda necesaria de un producto para un periodo determinado de un año a partir de la entrada de los últimos n periodos anteriores de ese año y años anteriores.

Primero hay que procesar las líneas de venta de cada uno de los albaranes para recuperar el histórico de la demanda de cada producto. Un albarán de venta es un documento mercantil que acredita la entrega de un pedido y donde aparece por líneas los productos o servicios vendidos. Un ejemplo de las líneas de un albarán sería el mostrado en la Tabla 4.3.

Con estas líneas de albarán se ha obtenido la demanda diaria de cada uno de los productos. Como los datos que se tenían iban desde 2014 a 2019 se consiguió la demanda de cada producto a lo largo de esos 2190 días. En total, entre las dos bases de datos, se obtuvo el consumo diario

Referencia	Descripción	Lote	Uds.	Cajas	Kilos	Descuento	Total
PO00122	Salchichón ibérico 225g	9123912	10	3.34	10	0	14.63
PO56172	Cinta de Lomo 1kg	2331223	3	3	3	0	18.43

Figura 4.3: Ejemplo de dos líneas de venta de un albarán de venta.

de 2200 productos que cumplieran una serie de requisitos con base a los factores analizados previamente:

- **Creación:** El producto debió de ser creado antes de 2014. Si el producto no fue creado antes de ese año los periodos que comenzarán desde el 2014 hasta la creación de ese producto, la demanda sería 0 y por tanto generaría ruido en los datos de entrenamiento.
- **Dado de baja:** El producto no ha sido dado de baja en ningún momento entre el año 2014 y 2019. Un producto dado de baja es aquel el cual no se vende bien porque su proveedor ya no lo distribuye o simplemente no daba el suficiente rendimiento. Este requisito hace que evitemos productos que se hayan dado de baja en un momento determinado y que generará una demanda de producto 0 en los periodos en los que estaba dado de baja.
- **Depósitos:** Todos los productos que se hayan adquirido en un momento determinado como depósito se eliminan. El día de compra del depósito contabiliza la venta de x unidades a ese cliente. El suministro posterior de las unidades se suministran a precio cero a ese cliente hasta completar el depósito. Es decir, es como hubiera ventas falsas a precio cero de ese producto. Al ser demasiada compleja la estructura de la base de datos para determinar cómo gestionar estos depósitos y ser una minoría de productos, me decante por despreciar simplemente éstos productos.
- **Rapples:** En principio, ninguna de las empresas que nos han proporcionado las base de datos utilizan rapples y por lo que no se han tenido en cuenta para el tratamiento de los datos.
- **Consumo:** El producto se debe haber consumido como mínimo en un 60% de los días. Este requisito hace que el conjunto de datos final tenga menos ruido, ya que existían casi 10000 productos con muy pocas ventas.

Una vez agrupado el consumo diario de cada uno los productos nos encontramos con un escenario como el de la Tabla 4.1. Cabe mencionar que la magnitud con el que se mide el consumo es con base en el formato de cantidad pedida de cada uno de los productos; es decir, un producto se puede vender por defecto por unidades, cajas, kilos o un formato personalizado. El formato que aparezca por defecto será la magnitud que represente el consumo de ese producto.

En este punto, quedan formar los periodos de demanda, pero, ¿cuánto tiempo es un periodo? ¿cómo se determina?. En este estudio probaremos varios valores para determinar un periodo de tiempo: una semana, semana y media, etc, como veremos más adelante. Para entender cómo se han agrupado los periodos, vamos a suponer que un periodo de demanda

Referencia	01/01/2014	02/01/2014	03/01/2014	...	31/12/2019
DAT1SP01041	2	5	3	...	0
DAT1SP01046	4	8	2	...	7

Tabla 4.1: Consumo diario de cada uno de los productos en su formato por defecto.

Referencia	t_1	t_2	t_3	...	t_{312}
1	22	11	21	...	14
2	35	31	25	...	23

Tabla 4.2: Consumo por periodos de cada uno de los productos en su formato por defecto.

es de una semana; si partimos de esa base, en total nos quedarían un total de 312 periodos de demanda para cada uno de los productos. Ver Tabla 4.2.

En este momento, solo falta matizar la estructura del conjunto de datos de los modelos. Por cada producto se crearán múltiples entradas para el conjunto de entrenamiento. Veamos un ejemplo de como se crearían las entradas. Imaginemos que queremos tener en cuenta los últimos 2 periodos; es decir, $n = 2$, y tenemos en cuenta los periodos del año pasado. El resultado sería que los productos crearían las entradas se muestra en la Tabla 4.3. Como se puede observar en esta Tabla, la primera entrada generada por el producto 1 tendría el siguiente desglose: t_1 , t_2 y t_3 que representarían la demanda histórica del año anterior, t_{53} y t_{54} representarían los n periodos anteriores al periodo a predecir y t_{55} sería el periodo a predecir, es decir, cada entrada constaría de un total de 5 valores. La segunda entrada generada por este producto seguiría la misma lógica sólo que desplazado un periodo a la derecha. De esta manera podemos concluir que si tenemos un total de 312 periodos obtendríamos un total de 260 entradas por cada producto.

Año	Y_1						Y_2					
Ref.	t_1	t_2	t_3	t_4	t_5	...	t_{53}	t_{54}	t_{55}	t_{56}	t_{57}	...
1	22	11	21			...	12	9	X			...
1		11	21	32		...		9	23	X		...
1			21	32	38	...			23	12	X	...
...
2	4	5	7			...	6	4	X			...
2		5	7	9		...		4	7	X		...
2			7	9	6	...			7	9	X	...

Tabla 4.3: Estructura de datos para el entrenamiento de los modelos.

Con esto, ya tendríamos el conjunto de datos para el entrenamiento del modelo; no obstante, hace falta definir el conjunto de datos de prueba. Éste es un conjunto de datos que lo utilizaremos para realizar la evaluación imparcial de los modelos. Para ello dividiremos el conjunto de datos generado en 88% para el conjunto de entrenamiento y un 12% para el conjunto de prueba.

4.3 Ajuste de parámetros en los modelos

En esta sección veremos los hiperparámetros que se han tenido en cuenta para cada uno de los modelos y los valores finales que se le han asignado a cada uno de ellos. Para establecer un valor a los hiperparámetros se ha utilizado una búsqueda aleatoria de hiperparámetros (Bergstra y Bengio, 2012), para esta búsqueda se muestran un número fijo de configuraciones de cada hiperparámetro del modelo a partir de unas distribuciones de probabilidad especificadas, además, esta búsqueda utilizará un método de validación cruzada para evitar el sobreajuste. A lo largo de esta sección, iremos viendo paso a paso cada una de las distribuciones de búsqueda elegidas y como ejemplo el valor obtenido de esta búsqueda para los datos filtros por los requisitos previamente explicados.

Todos los hiperparámetros que se van a mencionar en este apartado son los correspondientes a la librería `sklearn` que ha sido la utilizada para este proyecto (Buitinck y cols., 2013).

4.3.1 Árbol de regresión

Para este modelo se han tenido en cuenta los siguientes hiperparámetros:

- **max_depth**: La profundidad máxima del árbol. Rango de valores para la búsqueda: [5, 11].
- **min_samples_split**: El número mínimo de muestras necesarias para dividir un nodo interno. Rango de valores para la búsqueda: [5, 20].
- **min_samples_leaf**: El número mínimo de muestras necesarias para estar en un nodo hoja. Rango de valores para la búsqueda: [2, 20].

Hiperparámetros / Iteraciones	50	100	200
min_samples_split	9	13	13
min_samples_leaf	18	19	19
max_depth	8	6	6

Tabla 4.4: Resultado de la búsqueda aleatoria de hiperparámetros para el modelo de árbol de regresión.

4.3.2 Bosques extramadamente aleatorios

Para este modelo se han tenido en cuenta los siguientes hiperparámetros:

- **max_depth**: La profundidad máxima los árboles de regresión en el bosque. Rango de búsqueda: [6, 13].
- **min_samples_split**: El número mínimo de muestras necesarias para dividir un nodo interno de cada árbol en el bosque. Rango de búsqueda; [7, 16].
- **min_samples_leaf**: El número mínimo de muestras necesarias para estar en un nodo hoja de cada árbol en el bosque. Rango de búsqueda: [2, 13].

- **max_features**: La cantidad de características a considerar al buscar la mejor partición. Rango de búsqueda: [5, 13].
- **max_samples**: El número de muestras a extraer de valores del conjunto de datos para entrenar cada estimador base. Conjunto de valores para la búsqueda = { 0.7, 0.8, 0.9, 0.95, 1 }.
- **n_estimators**: El número de árboles en el bosque. Conjunto de valores de búsqueda = { 30, 40, 50, 70, 90, 100 }.

Hiperparámetros / Iteraciones	50	100	200
min_samples_split	11	7	7
min_samples_leaf	6	3	2
max_depth	11	10	12
max_samples	0.8	0.9	0.9
max_features	10	11	10
n_estimators	90	70	40

Tabla 4.5: Resultado de la búsqueda aleatoria de hiperparámetros para el modelo de bosques extremadamente aleatorios.

4.3.3 Adaboost con clasificador base de árboles de regresión

Para este modelo se han tenido en cuenta los siguientes hiperparámetros:

- **learning_rate**: Peso aplicado a cada clasificador en cada iteración de impulso. Conjunto de valores posibles para la búsqueda = { 0.005, 0.01, 0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35 }.
- **loss**: La función de pérdida que se usa al actualizar los pesos después de cada iteración de impulso. Conjunto de valores posibles para la búsqueda = { 'square', 'exponential', 'linear' }.

En este modelo, al utilizarse como un potenciador de rendimiento de los árboles de regresión, tiene que contemplar también los posibles hiperparámetros de los árboles de regresión esto se puede hacer en un bucle donde se van probando los diferentes valores para el clasificador débil al mismo tiempo que se hace la búsqueda aleatoria de hiperparámetros del Adaboost. Finalmente, la configuración del clasificador base y Adaboost que dé mejor resultado es el seleccionado.

Hiperparámetros / Iteraciones	5	10	20
learning_rate	0.25	0.1	0.01
loss	exponential	exponential	square

Tabla 4.6: Resultado de la búsqueda aleatoria de hiperparámetros para el modelo de Adaboost con clasificador base de árbol de regresión.

4.3.4 Extreme Gradient Boosting

Para este modelo se han tenido en cuenta los siguientes hiperparámetros:

- **max_depth**: Profundidad máxima de un árbol en el bosque. Conjunto de datos de búsqueda = { 5, 6, 7, 8, 9, 10 }.
- **learning_rate**: Reducción del tamaño del paso utilizada en la actualización para evitar el sobreajuste. Conjunto de valores posibles para la búsqueda = { 0.005, 0.01, 0.025, 0.05, 0.1, 0.15 }.
- **colsample_bynode**: Proporción de submuestra de columnas para cada partición. Conjunto de valores posibles para la búsqueda = { 0.5, 0.6, 0.7, 0.8, 0.9, 1.0 }.
- **colsample_bylevel**: Proporción de submuestras de columnas para cada nivel. Conjunto de valores posibles para la búsqueda = { 0.8, 0.9, 1.0 }.
- **colsample_bytree**: Proporción de submuestra de columnas al construir cada árbol. Conjunto de valores posibles para la búsqueda = { 0.6, 0.7, 0.8, 0.9, 1.0 }.
- **subsample**: Proporción de submuestras de las instancias de formación. Establecerlo en 0.5 significa que el modelo muestreará al azar la mitad de los datos de entrenamiento antes de crear árboles y esto evitará el sobreajuste. Conjunto de valores posibles para la búsqueda = { 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7 }.
- **min_child_weight**: Peso mínimo que necesita un árbol en el bosque para particionar. Conjunto de valores posibles para la búsqueda = { 5, 10, 15, 20, 25 }.
- **reg_alpha**: Plazo de regularización L1 sobre las ponderaciones. Incrementar este valor hará que el modelo sea más conservador. Conjunto de valores posibles para la búsqueda = { 1, 5, 10, 20, 50 }.
- **reg_lambda**: Plazo de regularización L2 sobre ponderaciones. Incrementar este valor hará que el modelo sea más conservador. Conjunto de valores posibles para la búsqueda = { 0.01, 0.05, 0.1, 0.5, 1 }.
- **n_estimators**: El número de árboles en el bosque. Conjunto de valores posibles para la búsqueda = { 100, 200, 500, 1000 }.

4.3.5 Perceptrón multicapa

Para este modelo se han tenido en cuenta los siguientes hiperparámetros:

- **hidden_layer_sizes**: Cantidad de perceptrones por cada capa oculta. Los valores de búsquedas utilizados son todas las combinaciones posibles entre un rango de capas [2, 7] y un rango de neuronas por cada capa [10, 60].
 - **alpha**: Parámetro de penalización L2. Conjunto de valores posibles para la búsqueda = { 5, 1, 0.5, 0.1, 0.05, 0.01, 0.001 }.
-

Hiperparámetros / Iteraciones	200	500	1000
max_depth	11	8	10
learning_rate	0.01	0.005	0.005
colsample_bynode	1	0.8	1
colsample_bylevel	1	1	0.8
colsample_bytree	0.7	0.8	1
subsample	0.5	0.6	0.6
min_child_weight	20	5	5
reg_alpha	10	20	50
reg_lambda	0.5	0.5	1
n_estimators	1000	1000	1000

Tabla 4.7: Resultado de la búsqueda aleatoria de hiperparámetros para el modelo Extreme Gradient Bosting.

- **learning_rate_init:** Tasa de aprendizaje inicial utilizada. Controla el tamaño del paso al actualizar los pesos. Conjunto de valores posibles para la búsqueda = { 0.05, 0.01, 0.005, 0.001, 0.0005 }.
- **beta_1:** Tasa de desintegración exponencial para las estimaciones del vector de primer momento en adam (Kingma y Ba, 2014).
- **beta_2:** Tasa de decaimiento exponencial para las estimaciones del segundo vector de momento en adam (Kingma y Ba, 2014).

Hiperparámetros / Iteraciones	10	20	50
hidden_layer_sizes	{ 50, 50, 50, 50 }	{ 50, 50, 50, 50, 50 }	{ 50, 50, 50 }
alpha	1	0.1	0.001
learning_rate_init	0.01	0.001	0.0005
beta_1	0.975	0.995	0.995
beta_2	0.99	0.999	0.999

Tabla 4.8: Resultado de la búsqueda aleatoria de hiperparámetros para el modelo de perceptrón multicapa.

5 Experimentación y resultados

Un punto importante en la experimentación de los modelos es cuantificar la precisión de estos. Para ello, se utilizan diferentes indicadores que nos dan una idea; no obstante, cada indicador tiene sus ventajas y desventajas y no todos sirven para lo mismo. Para este estudio se han elegido principalmente tres indicadores que nos van a proporcionar una idea de la precisión de los modelos:

- **Bias:** El sesgo (Bias) de pronóstico es el promedio de error y se define como:

$$Bias = \frac{1}{n} \sum_{i=1}^n (f_i - t_i) \quad (5.1)$$

Un sesgo de pronóstico ocurre cuando existen diferencias consistentes entre los resultados reales y los pronósticos generados previamente de esas cantidades; es decir, los pronósticos pueden tener una tendencia general a ser demasiado altos o demasiado bajos. Esto es debido a que un error positivo en un período puede compensar un error negativo en otro período. En este contexto un modelo puede lograr un sesgo muy bajo y no ser preciso. Sin embargo, un pronóstico muy sesgado es un indicador de que hay algún tipo problema en el modelo.

- **MAE:** El error absoluto medio (MAE) es la media del error absoluto:

$$MAE = \frac{1}{n} \sum_{i=1}^n |f_i - t_i| \quad (5.2)$$

Al contrario que el Bias, el MAE es un valor absoluto. Si MAE es 10 para un producto en particular, no puede saber si esto es bueno o malo. Si la demanda promedio es 1.000, este valor es, por supuesto, bueno, pero si la demanda promedio es 1, un MAE de 10 indica que la precisión del modelo es muy deficiente. Para solucionar esto, es común dividir MAE por la demanda promedio para obtener un porcentaje escalado que nos dicte la distancia promedio en la que el pronóstico se aleja del valor real. En este estudio utilizaremos el MAE dividido por la demanda promedio al que denominaremos como MAE%.

- **RMSE:** El error cuadrático medio (RMSE) se define como la raíz cuadrada del error de pronóstico cuadrático medio:

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n \left(\frac{f_i - t_i}{\sigma_i} \right)^2} \quad (5.3)$$

Esta, asigna más peso a los errores más grandes. Esto es interesante de cara a los valores atípicos pues podría darse el caso de tener un MAE relativamente bajo, pero esto no nos asegura que esté prediciendo correctamente, ya que puede ser que las atipicidades directamente no las esté teniendo en cuenta; no obstante, este indicador es una medida de lo bien que el modelo está tratando los valores atípicos. Al igual que pasa con el MAE, en este estudio, dividiremos el valor del RMSE entre la demanda promedio para tener un porcentaje escalado e interpretable al que llamaremos RMSE%.

Una vez definidos los indicadores vamos a ver los datos que se utilizarán para el entrenamiento. Entrenaremos los modelos con dos bases de datos diferentes. Estas dos bases de datos tienen diferencias pues se tratan de dos empresas distintas. Cada empresa trabaja de una manera, venden productos de familias totalmente diferentes y también se encuentran en zonas geográficas diferentes. En este contexto, el desglose de los datos podemos verlo en la Tabla 5.1 teniendo en cuenta:

- **Demanda histórica formada con el consumo de productos desde 2014 hasta 2020:** No se ha tenido en cuenta el año 2020 ni posteriores debido a la crisis sanitaria. Se probó a incorporar hasta 2021, pero los resultados fueron menos precisos debido al ruido que generó este año.
- **Periodos de tiempo:** Se usaron finalmente periodos de una semana. Se probaron tres diferentes periodos de tiempo: un día, 3 días, una semana y una semana y media. La que mejor resultados proporcionó con una diferencia del 12% respecto al segundo, fue el periodo de una semana.
- **Periodos anteriores:** Los periodos anteriores que se han tenido en cuenta son: 5 periodos de hace 2 años desde $i - 109$ hasta $i - 103$; 5 periodos del año anterior empezando en $i - 57$ y terminando en $i - 51$; y, por último, 4 periodos anteriores al periodo a predecir, es decir, desde $i - 5$ hasta i . En total cada entrada de datos tendrá un total de 14 elementos y el conjunto de salida un elemento i que será el periodo a predecir.

En la Tabla 5.1 "con filtros" se refiere al conjunto de datos una vez aplicados los requisitos explicados en el Capítulo 4: Procesamiento y estructura.

Conjunto	Productos	Entradas generadas	Train (88%)	Test (12%)
Empresa 1	1834	572155	510753	61402
Empresa 1 con filtros	135	42230	37698	4532
Empresa 2	3898	1216470	1085922	130548
Empresa 2 con filtros	1310	408770	364902	43868

Tabla 5.1: Resultados de los indicadores seleccionados aplicando un árbol de regresión en los diferentes escenarios.

Para la experimentación de los modelos se han realizado varios escenarios que se aplicarán a cada uno de los modelos para estudiar posteriormente sus resultados:

- **Escenario 1:** Con la base de datos de la empresa 1.

- A: Todas las ventas sin filtros.
 - B: Con filtros.
 - C: Con filtros y sólo los productos con un consumo de media de 50.
 - D: Con filtros y sólo los productos con un consumo con media inferior a 50.
- **Escenario 2:** Con la base de datos de la empresa 2.
 - A: Todas las ventas sin filtros.
 - B: Con filtros.
 - C: Con filtros y sólo los productos con un consumo de media de 50.
 - D: Con filtros y sólo los productos con un consumo con media inferior a 50.
- **Escenario 3:** Las dos base de datos.
 - A: Todas las ventas sin filtros.
 - B: Con filtros.
 - C: Con filtros y sólo los productos con un consumo de media de 50.
 - D: Con filtros y sólo los productos con un consumo con media inferior a 50.

5.1 Resultados de los modelos

Antes de entrenar un modelo se realiza una búsqueda aleatoria de hiperparámetros para determinar cuáles serán los mejores parámetros para el conjunto de datos de entrada. Esto es debido a que cada punto de los escenarios utiliza un conjunto de datos con unas características diferentes. Posteriormente a la búsqueda, es cuando se ha entrenado y evaluado el modelo y los resultados expuestos en su correspondiente tabla.

5.1.1 Árbol de regresión

Los resultados presentados en los diferentes escenarios se pueden ver en la Tabla 5.2. Más adelante analizaremos estos resultados con los de los otros modelos. En las Figuras 5.1 y 5.2 se muestran los periodos predichos y su demanda real en diferentes productos y escenarios.

5.1.2 Bosque extremadamente aleatorio

Los resultados presentados en los diferentes escenarios se pueden ver en la Tabla 5.3. En las Figuras 5.3 y 5.4 se muestran los periodos predichos y su demanda real en diferentes productos y escenarios.

5.1.3 Adaboost con árboles de regresión como clasificadores base

Los resultados presentados en los diferentes escenarios se pueden ver en la Tabla 5.4. En las Figuras 5.5 y 5.6 se muestran los periodos predichos y su demanda real en diferentes productos y escenarios.

		MAE%		RMSE%		Bias%	
		Train	Test	Train	Test	Train	Test
Escenario 1	A	81.2	90	1154.0	1286.1	0.0	-0.7
	B	42.9	48	137.0	141.0	0.0	-3.2
	C	35.6	46.3	56.4	76.9	0.0	-4.2
	D	17.7	25.1	27.7	36.5	0.0	-3.4
Escenario 2	A	67.4	67.6	487.5	429.6	0.0	1.8
	B	43.0	48.1	96.8	110.0	0.0	0.1
	C	39.4	45.1	73.1	45.1	0.0	2.4
	D	26.7	33.0	43.3	53.5	0.0	-1.6
Escenario 3	A	78.1	83.4	940.3	1047.8	0.0	-1.5
	B	57.8	61.0	309.5	337.1	0.0	-0.9
	C	47.6	53.1	90.0	101.8	0.0	-2.5
	D	25.7	30.9	41.6	49.2	0.0	-3.4

Tabla 5.2: Resultados de los indicadores seleccionados aplicando un árbol de regresión en los diferentes escenarios.

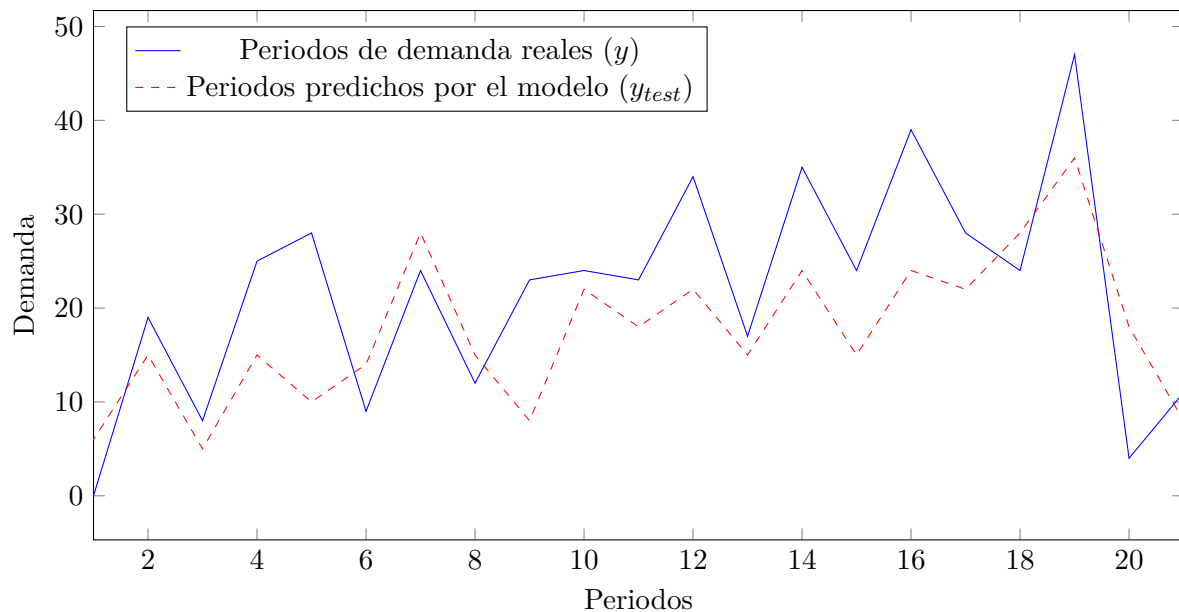


Figura 5.1: Comparación de los valores reales de demanda de un producto con los predichos por el modelo del árbol de regresión para el Escenario 1 B.

5.1.4 Extreme Gradient Boosting

Los resultados presentados en los diferentes escenarios se pueden ver en la Tabla 5.5. En las Figuras 5.7 y 5.8 se muestran los periodos predichos y su demanda real en diferentes productos y escenarios.

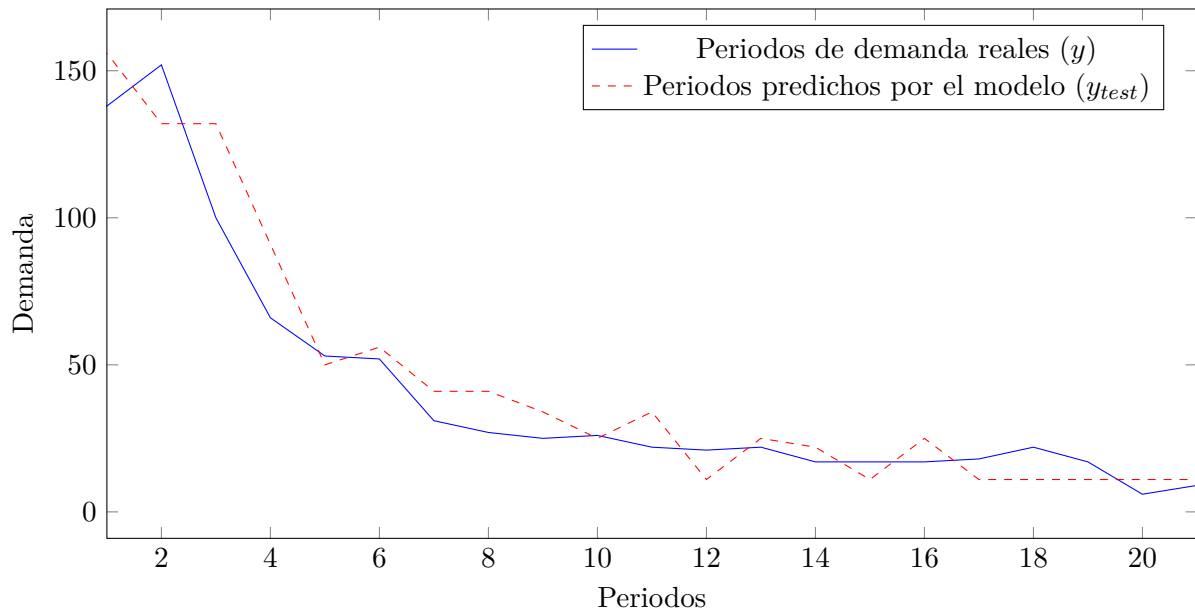


Figura 5.2: Comparación de los valores reales de demanda de un producto con los predichos por el modelo del árbol de regresión para el Escenario 2 B.

		MAE%		RMSE%		Bias%	
		Train	Test	Train	Test	Train	Test
Escenario 1	A	67.8	80.9	931.9	1203.7	0.3	-0.2
	B	38.6	45.8	122.1	134.6	0.1	-4.6
	C	28.9	41.7	47.4	65.1	0.1	-4.7
	D	14.8	24.4	24.5	40.1	0.0	-5.8
Escenario 2	A	61.6	62.8	451.0	399.7	0.0	2.4
	B	50.2	54.0	267.9	306.8	0.0	-0.1
	C	32.6	42.8	65.9	87.1	-0.1	2.7
	D	20.7	27.3	34.8	43.7	0.0	-0.3
Escenario 3	A	68.6	73.5	787.6	956.4	0.1	1.0
	B	73.5	61.0	309.5	337.1	0.0	-0.9
	C	33.6	42.2	64.5	81.5	0.0	-0.8
	D	18.2	25.7	31.3	42.2	-0.1	-1.2

Tabla 5.3: Resultados de los indicadores seleccionados aplicando bosques extremadamente aleatorios en los diferentes escenarios.

5.1.5 Perceptrón Multicapa

Los resultados presentados en los diferentes escenarios se pueden ver en la Tabla 5.6. En las Figuras 5.9 y 5.10 se muestran los periodos predichos y su demanda real en diferentes productos y escenarios.

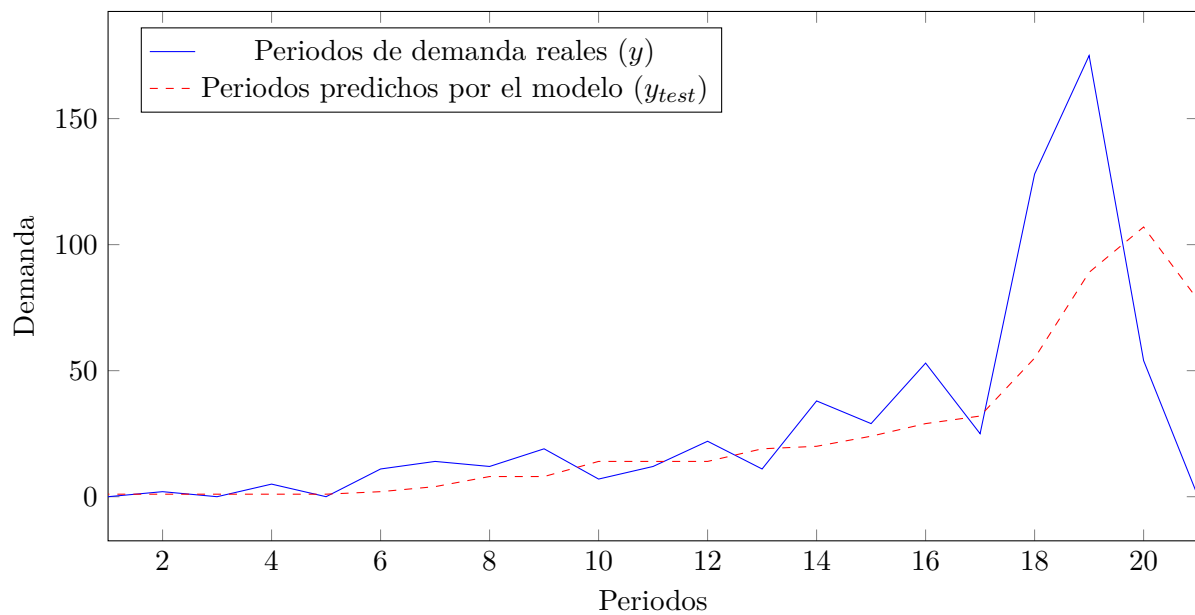


Figura 5.3: Comparación de los valores reales de demanda de un producto con los predichos por el modelo de bosque extremadamente aleatorio para el Escenario 1 B.

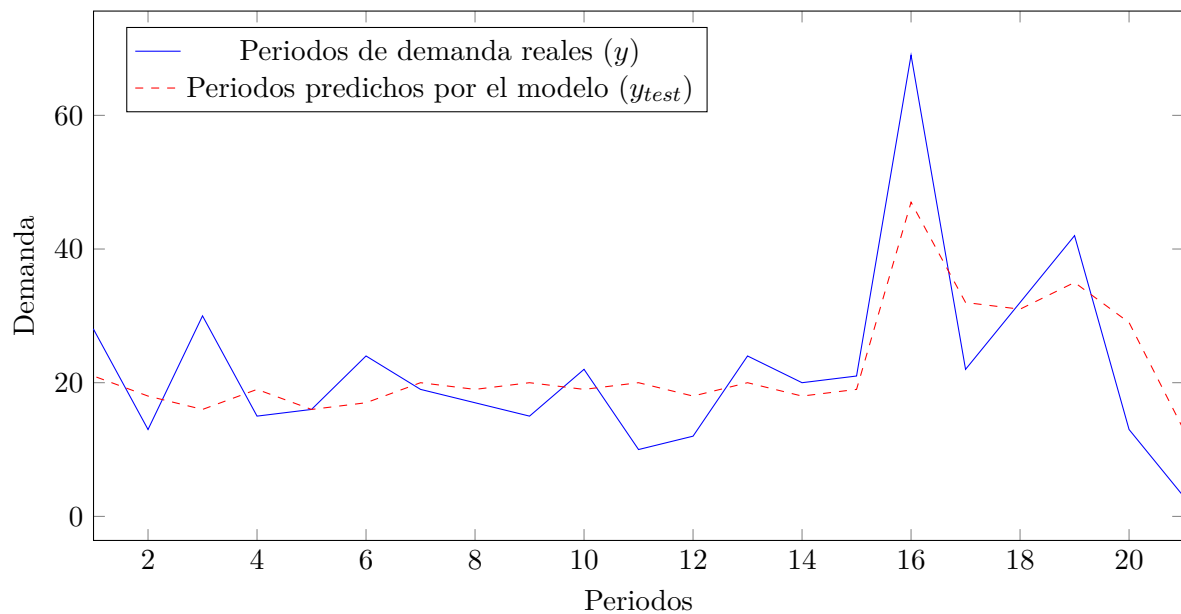


Figura 5.4: Comparación de los valores reales de demanda de un producto con los predichos por el modelo de bosque extremadamente aleatorio para el Escenario 2 B.

		MAE%		RMSE%		Bias%	
		Train	Test	Train	Test	Train	Test
Escenario 1	A	33.9	77.3	272.5	1491.8	2.6	0.1
	B	19.6	43.9	45.8	139.4	1.9	4.5
	C	29.9	42.0	30.2	71.4	0.7	-0.6
	D	7.2	24.4	10.2	40.1	0.1	0.7
Escenario 2	A	51.6	61.5	358.9	400.8	-3.1	2.4
	B	38.6	54.1	200.0	343.3	0.1	6.7
	C	16.7	40.4	25.0	84.5	0.6	10.1
	D	8.6	26.2	13.1	48.8	0.	4.8
Escenario 3	A	52.5	62.1	361.3	402.3	-3.9	2.1
	B	37.0	51.5	175.0	288.4	0.5	5.5
	C	23.5	40.3	33.7	79.6	0.9	6.0
	D	7.4	25.7	12.4	42.6	0.3	4.2

Tabla 5.4: Resultados de los indicadores seleccionados aplicando Adaboost con árboles de regresión como clasificadores base en los diferentes escenarios.

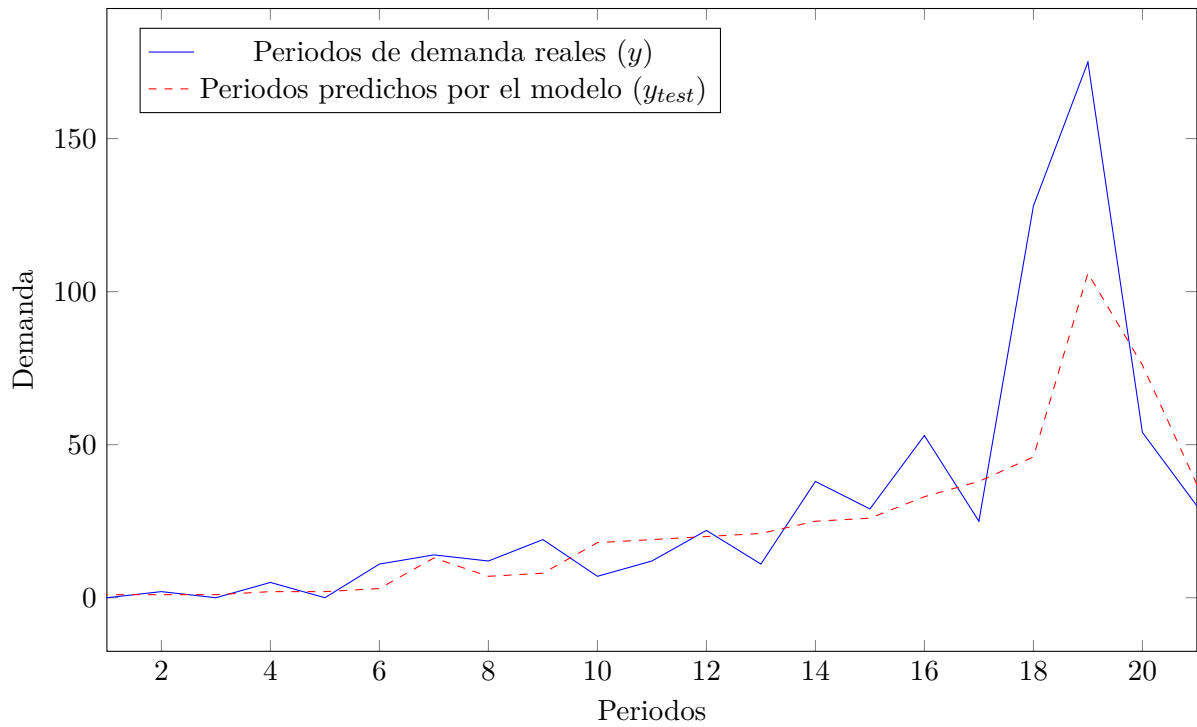


Figura 5.5: Comparación de los valores reales de demanda de un producto con los predichos por el modelo de Adaboost con árboles de regresión como clasificadores base para el Escenario 1 B.

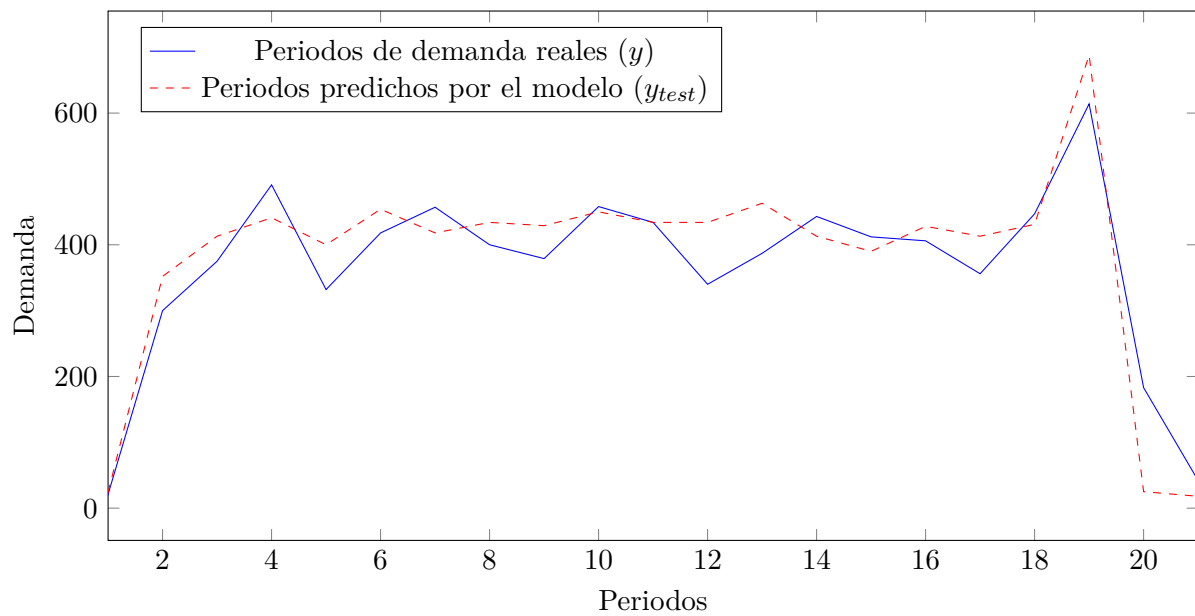


Figura 5.6: Comparación de los valores reales de demanda de un producto con los predichos por el modelo de Extreme Gradient Boosting para el Escenario 2 B.

		MAE%		RMSE%		Bias%	
		Train	Test	Train	Test	Train	Test
Escenario 1	A	66.7	76.1	1076.0	1210.3	14.9	11.5
	B	38.1	44.3	130.9	133.5	3.8	0.6
	C	31.9	41.0	51.2	67.1	8.5	5.7
	D	23.0	23.6	52.7	46.0	13.3	5.4
Escenario 2	A	55.0	58.0	436.9	384.4	-1.4	3.8
	B	47.4	52.0	292.1	326.9	2.1	3.7
	C	31.4	40.0	63.8	80.8	3.5	6.0
	D	16.4	25.5	23.9	40.1	0.3	1.7
Escenario 3	A	62.2	68.2	835.9	964.3	-6.9	5.9
	B	45.8	50.2	257.9	287.3	2.5	3.3
	C	32.6	40.7	61.1	78.7	5.2	6.2
	D	18.1	25.0	27	41.0	0.8	0.5

Tabla 5.5: Resultados de los indicadores seleccionados aplicando el modelo Extreme Gradient Boosting en los diferentes escenarios.

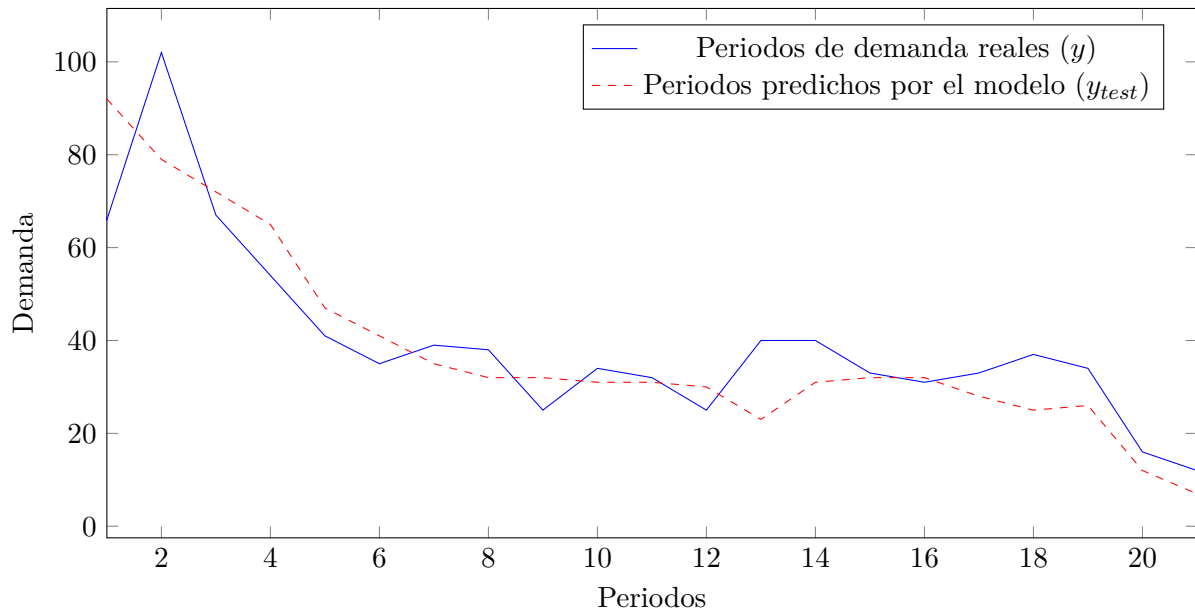


Figura 5.7: Comparación de los valores reales de demanda de un producto con los predichos por el modelo de Extreme Gradient Extreme para el Escenario 1 B.

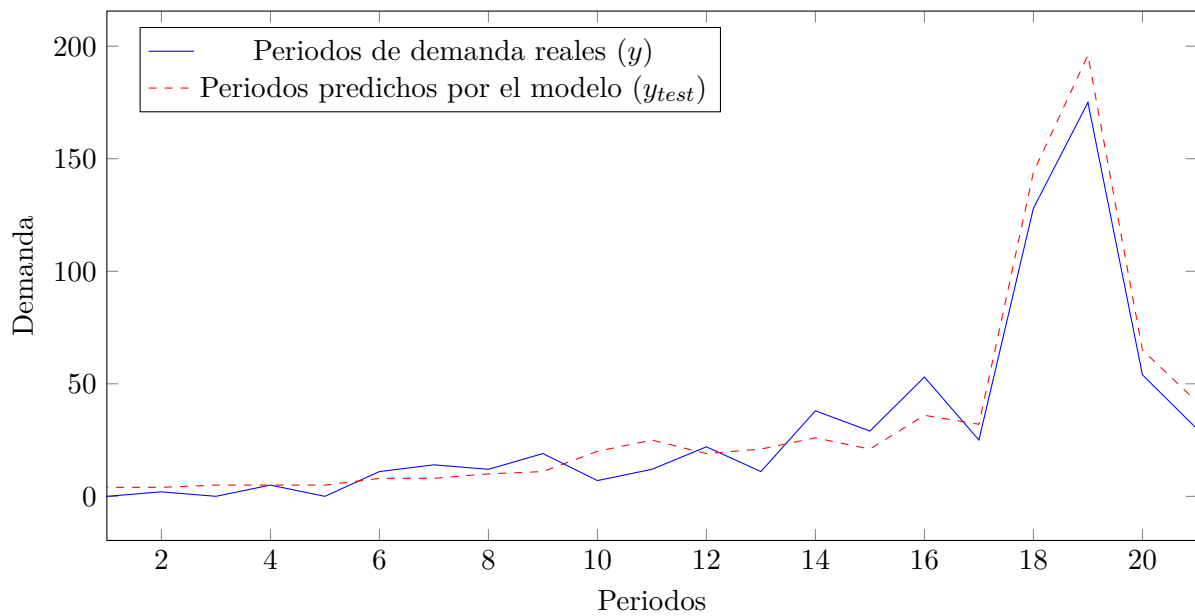


Figura 5.8: Comparación de los valores reales de demanda de un producto con los predichos por el modelo de Extreme Gradient Extreme para el Escenario 2 B.

		MAE%		RMSE%		Bias%	
		Train	Test	Train	Test	Train	Test
Escenario 1	A	64.5	78.1	955.3	1179.2	7.1	-0.4
	B	38.0	43.7	128.3	133.1	6.3	3.3
	C	33.1	41.1	56.4	66.8	12.2	8.9
	D	14.8	22.1	24.3	36.4	1.5	-0.1
Escenario 2	A	62.9	61.2	480.5	381.6	14.0	11.4
	B	59.1	60.6	335.4	323.2	-2.5	-2.6
	C	34.3	43.9	52.8	95.0	11.9	16.9
	D	20.1	25.4	29.7	41.6	0.6	3.9
Escenario 3	A	91.5	88.2	959.1	937.4	-16.8	-14.0
	B	52.7	53.2	302.4	277.6	5.9	6.5
	C	35.1	44.3	55.0	92.0	10.9	9.9
	D	22.5	24.7	36.1	40.6	0.6	0.8

Tabla 5.6: Resultados de los indicadores seleccionados aplicando el modelo perceptrón multicapa en los diferentes escenarios.

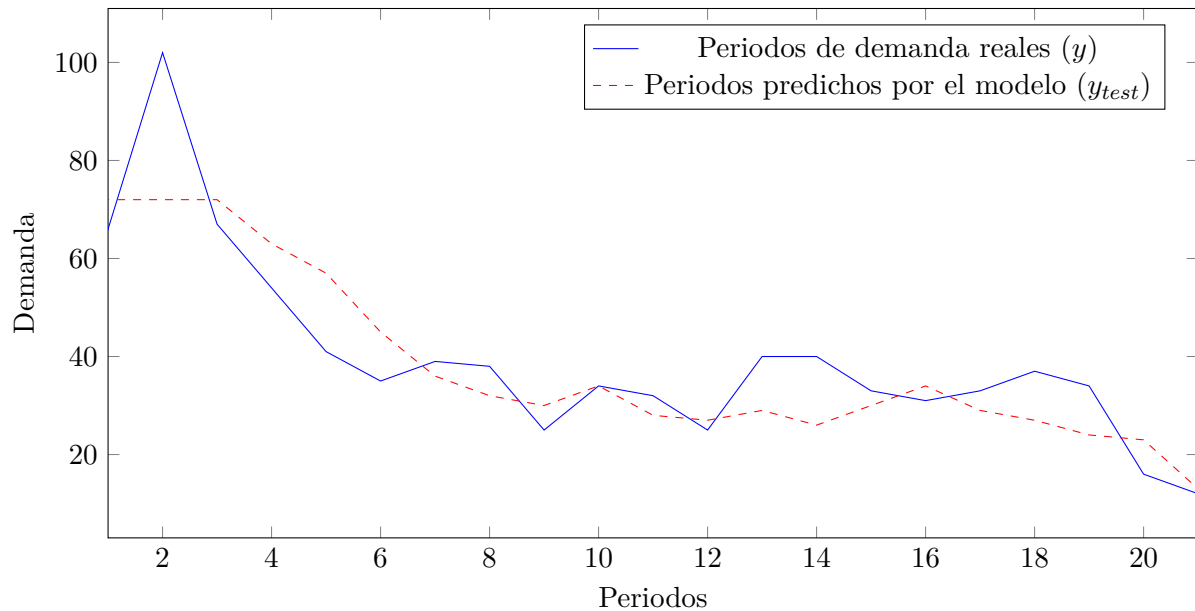


Figura 5.9: Comparación de los valores reales de demanda de un producto con los predichos por el modelo perceptrón multicapa para el Escenario 1 B.

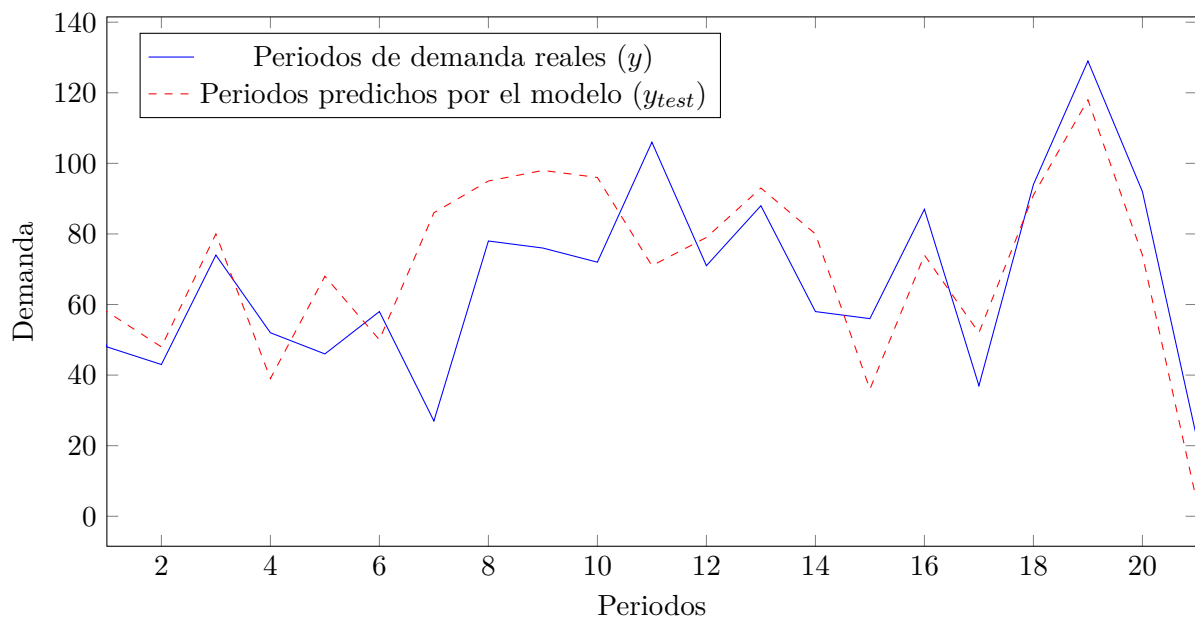


Figura 5.10: Comparación de los valores reales de demanda de un producto con los predichos por el modelo perceptrón multicapa para el Escenario 2 B.

5.2 Análisis de los resultados obtenidos

Como se ha podido observar en las tablas resultantes anteriores, los datos sin filtrar presentan unos resultados muy deficientes, esto, es debido a todos los factores que se explicaron previamente. Por otro lado, se puede observar como todos los modelos mejoran cuando se utilizan los datos filtrados de manera considerable.

Otro punto importante es cómo mejoran los modelos cuando se entrenan segmentando los datos en función del promedio de los periodos. Como se puede observar, por ejemplo, en la Tabla 5.2 (Escenario 1), el modelo entrena mejor separando los productos de menor a mayor en función de las cantidades vendidas, ¿por qué ocurre esto?. Esto ocurre debido a las atipicidades. Cuando se vende en menor cantidades la desviación estándar es menor y, por tanto, es más fácil de superar. Por ejemplo, si un producto tiene una desviación típica de 2 unidades es muy factible que de repente surja una compra de 5 unidades, ya que en términos de cantidad no es nada desorbitado y esto produce una atipicidad en la demanda. Es por ello, que en el Escenario 1 D el modelo que se ha entrenado sólo con productos con un promedio de consumo mayor a 50 funciona tan bien. Esto es debido a que es más complicado superar la desviación estándar consiguiendo que este tenga menos atipicidades.

Por otro lado, podemos ver el rendimiento obtenido por los modelos tratando las bases de datos de manera individual y conjuntamente. Cuando se tratan los datos en conjunto ambas bases de datos, la precisión de los modelos es mejor que tratando individualmente la primera base de datos, aunque generalmente mejora los de la segunda. Para poder llegar a una respuesta con fundamento a esto se necesitarían más bases de datos de diferentes empresas. Una hipótesis a plantear es el papel que juegan factores tales como que cada empresa distribuya diferentes productos, estén en distintas zonas geográficas y tengan una organización diferente de su trabajo. No obstante, para poder evaluar rigurosamente el papel cada uno de estos factores se necesitarían más datos.

A continuación, vamos a realizar un análisis de los diferentes modelos:

- **Árbol de regresión:** Como se puede observar presenta un Bias% bajo tanto en el entrenamiento como en la evaluación. Ésto se debe a que los hiperparámetros seleccionados hacen que el árbol sea profundo y se ajuste bien a los datos en el entrenamiento, pero manteniendo buenos resultados en la dispersión. Sin embargo, los resultados que arroja de manera general no son muy buenos, ya que, como se puede observar en la Tabla 5.2, los resultados de MAE% en el conjunto de evaluación son altos. Por ejemplo, el Escenario 3 D nos indica que el porcentaje en promedio de error entre el valor real y el valor predicho es de un 30.9%.
- **Bosques extremadamente aleatorios:** Presentan un Bias% mayor que los árboles de decisión, aunque mejores resultados. Como se puede observar en la Tabla 5.3 el Escenario 3D de este modelo presenta una mejoría del 5,2% en MAE% y un 7% en RMSE respecto a un simple árbol de regresión. Esto demuestra que este modelo es más robusto y que las decisiones de un conjunto de árboles funcionan mejor que de uno solo.
- **Adaboost:** Este modelo parece presentar un caso de sobreajuste. Como se puede comprobar en la Tabla 5.4 ,se presentan unos resultados, por lo general, excelentes en el entrenamiento, pero bastante peores en la evaluación. En el conjunto de entrenamiento

presenta mejores resultados que los árboles extremadamente aleatorios, y en la evaluación, los mismos resultados en el MAE% y peores resultados en el control de las atipicidades según el indicador RMSE%.

- **Extreme Gradient Boosting:** En este caso nos encontramos ante un modelo que presenta un Bias% excesivamente alto en los conjuntos de datos con más atipicidades como pueden ser el Escenario 1 C, 2 C y 3 C. En cambio, como se muestra en la Tabla 5.5, ante conjuntos de datos con pocas atipicidades presenta mejores resultados que los anteriores modelos además de tener un MAE%, RMSE% y un Bias% equilibrado.
- **Perceptrón multicapa:** Este modelo sigue el mismo patrón que el modelo anterior: alto sesgo cuando se enfrenta contra conjuntos de datos con muchas atipicidades, pero excelente rendimiento en la evaluación para aquellos que no lo presentan, siendo el mejor en este último.

Por último, vamos a probar los diferentes modelos ante el producto definido en la Tabla 5.7. Se puede ver como este muestra grandes atipicidades y bruscos cambios de tendencias en la Figura 5.11. Partimos de que los modelos se van a entrenar con el conjunto de datos del Escenario 3 D, con periodos semanales y teniendo en cuenta 2 años atrás. El objetivo de los modelos será predecir los periodos finales de 2016. El resultado se muestra en 5.12.

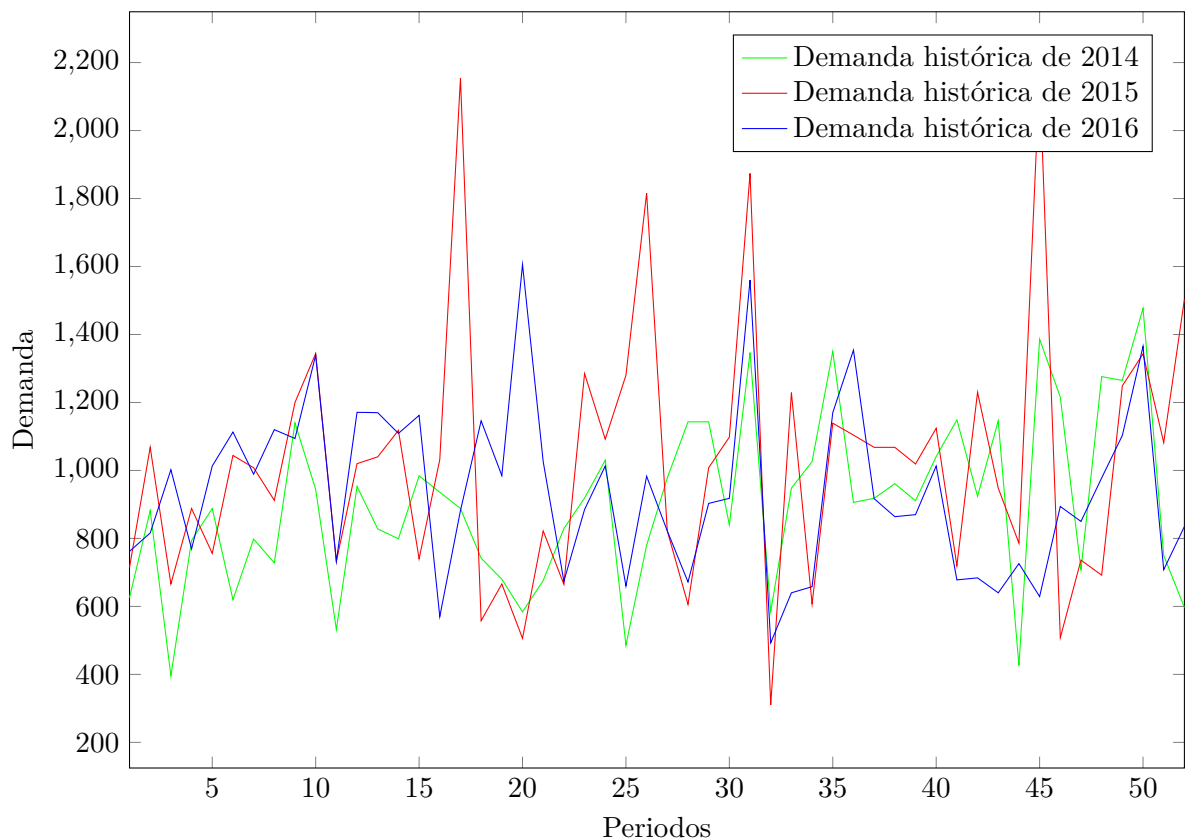


Figura 5.11: Demanda histórica de producto con un promedio de consumo de más de 50.

	2014					2015					2016				
	X_{11}	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}	X_{11}	X_{12}	X_{13}	X_{14}	Y
f_1	127	135	110	112	176	72	53	110	160	186	82	67	90	92	156
f_2	135	110	112	176	20	53	110	160	186	32	67	90	92	156	49
f_3	110	112	176	20	103	110	160	186	32	101	90	92	156	49	64
f_4	112	176	20	103	90	160	186	32	101	93	92	156	49	64	66
f_5	176	20	103	90	113	186	32	101	93	108	156	49	64	66	117
f_6	20	103	90	113	101	32	101	93	108	75	49	64	66	117	135
f_7	103	90	113	101	97	101	93	108	75	152	64	66	117	135	92
f_8	90	113	101	97	131	93	108	75	152	98	66	117	135	92	86
f_9	113	101	97	131	141	108	75	152	98	133	117	135	92	86	87
f_{10}	101	97	131	141	59	75	152	98	133	72	135	92	86	87	101
f_{11}	97	131	141	59	97	152	98	133	72	110	92	86	87	101	68
f_{12}	131	141	59	97	113	98	133	72	110	118	86	87	101	68	68
f_{13}	141	59	97	113	92	133	72	110	118	115	87	101	68	68	64
f_{14}	59	97	113	92	116	72	110	118	115	100	101	68	68	64	73
f_{15}	97	113	92	116	132	110	118	115	100	81	68	68	64	73	63
f_{16}	113	92	116	132	120	118	115	100	81	129	68	64	73	63	89
f_{17}	92	116	132	120	133	115	100	81	129	176	64	73	63	89	85
f_{18}	116	132	120	133	95	100	81	129	176	72	73	63	89	85	98
f_{19}	132	120	133	95	146	81	129	176	72	145	63	89	85	98	110
f_{20}	120	133	95	146	160	129	176	72	145	136	89	85	98	110	136
f_{21}	133	95	146	160	167	176	72	145	136	157	85	98	110	136	71
f_{22}	95	146	160	167	19	72	145	136	157	24	98	110	136	71	84

Tabla 5.7: Entradas y salidas de un producto de con un promedio de consumo de más de 50.

Como conclusión final, el modelo que ha presentado un rendimiento en los indicadores de forma más equilibrada y que en la práctica ha funcionado mejor ha sido Extreme Gradient Boosting.

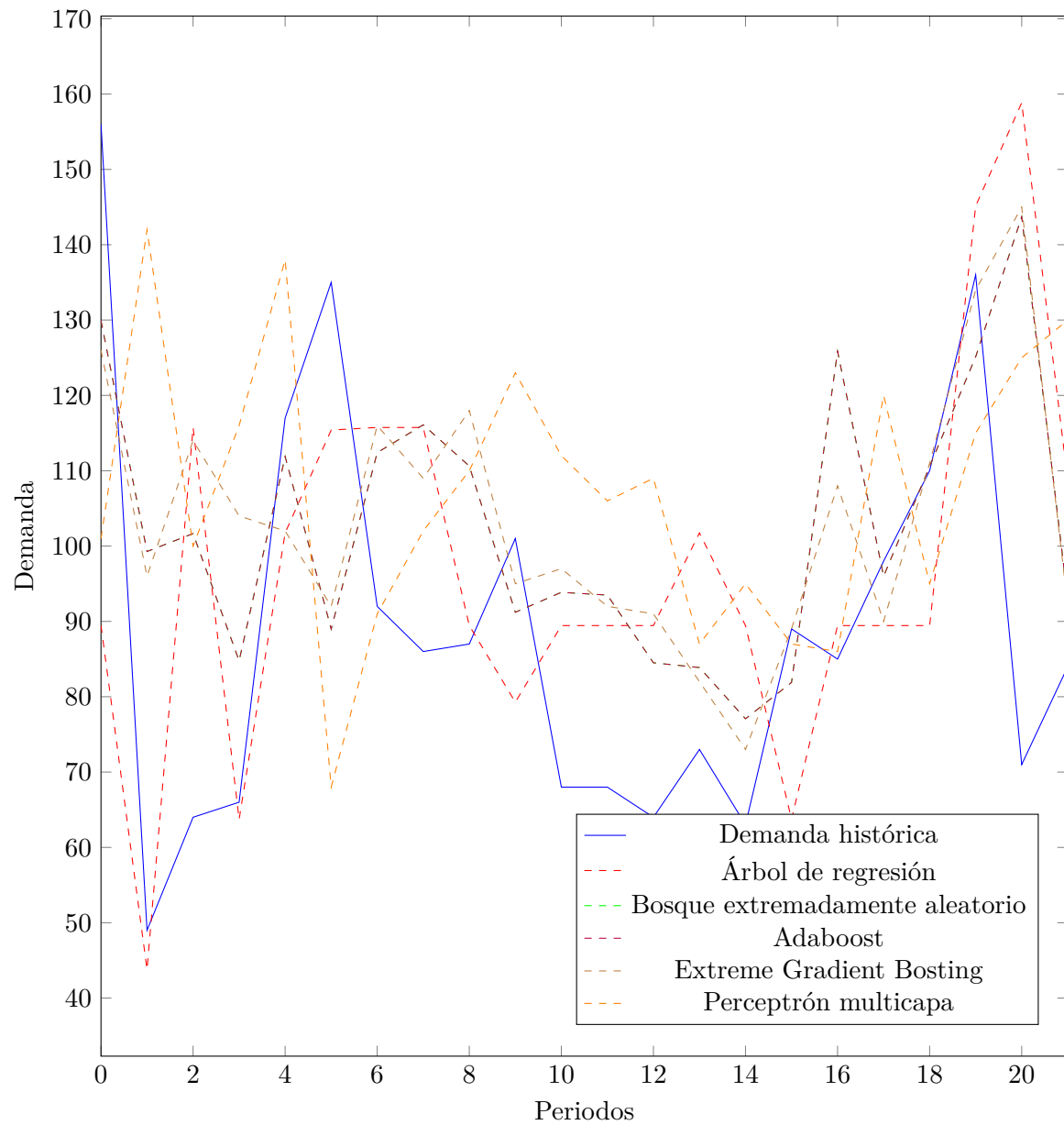


Figura 5.12: Demanda histórica de producto con un promedio de consumo de más de 50.

6 Conclusiones y trabajos futuros

Durante este trabajo se han visto como han evolucionado los modelos estadísticos utilizados hasta el momento por las empresas de distribución de alimentación y bebidas. Hemos visto como esos modelos presentan sus ventajas y sus desventajas. Posteriormente, hemos explicado el funcionamiento de nuevas técnicas de Inteligencia Artificial que pueden mirar patrones muchos más complejos que los modelos estadísticos pudiendo ser posibles sustitutos. Más adelante, se ha explicado como se calcula realmente la previsión final de stock necesaria en un periodo en este tipo de empresas y analizado los factores que le afectan. Se ha determinado que no es un problema fácil de manejar debido a la gran cantidad de variables a tener en cuenta y por ello, solo en este trabajo se han estudiado los factores internos que lo afectan. Más adelante, se ha explicado la gestión llevada de los datos y como se han formalizado las entradas de datos para el entrenamiento de los modelos. Finalmente, se han presentado una serie de escenarios que nos planteaban diferentes perspectivas del problema y analizando el rendimiento de los modelos en base a los escenarios. El análisis de la precisión de los resultados de los modelos investigados usando diferentes indicadores (BIAS, MAE y RMSE) muestran que los modelos con mejor rendimiento en la predicción de stock son el Extreme Gradient Boosting (XGB) y el Perceptrón Multicapa. El modelo XGB presenta un sesgo más elevado cuando se enfrenta a conjuntos de datos con muchas atipicidades, pero muestra un excelente rendimiento en relación a los otros modelos de aprendizaje automático. El Perceptrón Multicapa presenta, asimismo, un alto sesgo en conjuntos de datos con muchas atipicidades, pero muestra un excelente rendimiento en las entradas que no poseen tantas atipicidades. Estos dos modelos han sido los mejores en la predicción de stock de estas empresas con los datos proporcionados. Por otra parte, todos los modelos han presentado mejores resultados tratando los datos en dos grupos: productos con una demanda media inferior a 50 y con una superior a 50. Se concluyó que esto, es debido a que cuando el consumo de un producto es inferior es más fácil que surjan compras que superen los límites de valores típicos y por ende se creen más periodos atípicos haciendo que estos den peores resultados que los de un consumo elevado, ya que estos no poseen tantas atipicidades.

Como hemos visto en este trabajo, los modelos de inteligencia artificial poseen un gran potencial para la predicción de stock en las empresas del sector de la distribución de alimentos y bebidas. Aunque en nuestro estudio estos modelos tienen un gran margen de error, los resultados de la predicción de stock son superiores a los obtenidos por los modelos estadísticos utilizados normalmente en los softwares EPR del sector. Por ello, en el futuro, los modelos de inteligencia artificial deberán desarrollarse y mejorarse para que puedan realmente aplicarse en las empresas del sector. A corto plazo, sería interesante aplicar otros modelos de inteligencia artificial basados en redes neuronales como las redes recurrentes LSTM. Las redes LSTM suelen tener un rendimiento excelente en la predicción de series temporales que es, al fin y al cabo, el objetivo final de la predicción de stock en las empresas de distribución de alimentos y bebidas. Por otra parte, para evaluar su rendimiento y potencial real, sería necesario aplicar

los modelos a empresas del sector que distribuyan distintos tipos de productos. Una línea de desarrollo a largo plazo será la de analizar e incorporar en estos modelos de predicción de stock los factores externos; particularmente, el Índice de Precios de Consumo (IPC). Una forma de considerar los factores externos sería la de estudiar posibles correlaciones entre el conjunto de datos y los factores externos para posteriormente incorporarlos como entradas en los modelos. De esta forma, se podría mejorar la precisión de los modelos y conseguir resultados más robustos y precisos.

Bibliografía

- Bergstra, J., y Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2).
- Brown, R. G. (1956). Exponential smoothing for predicting demand. cambridge, mass., arthur d. little. *Book Exponential Smoothing for Predicting Demand*.
- Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., ... Varoquaux, G. (2013). API design for machine learning software: experiences from the scikit-learn project. En *Ecml pkdd workshop: Languages for data mining and machine learning* (pp. 108–122).
- Burns, A. F., y Mitchell, W. C. (1946). *Measuring business cycles* (no burn46-1). National bureau of economic research.
- Cam, A., Hall, B., y Chui, M. (2021). *Global ai survey: Ai proves its worth, but few scale impact*. <https://www.mckinsey.com/featured-insights/artificial-intelligence/global-ai-survey-ai-proves-its-worth-but-few-scale-impact>. ([Online; accessed 28-Mayo-2021])
- Cartas, A. (s.f.). *Diagrama de perceptrón*. <https://commons.wikimedia.org/w/index.php?curid=41534843>. ([Online; accessed 31-Mayo-2021])
- Cheng, L. (2019). *Basic ensemble learning (random forest, adaboost, gradient boosting)- step by step explained*. <https://towardsdatascience.com/basic-ensemble-learning-random-forest-adaboost-gradient-boosting-step-by-step-explained-95d49d1e2725>. ([Online; accessed 30-Mayo-2021])
- Freund, Y., Schapire, R., y Abe, N. (1999). A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, 14(771-780), 1612.
- Freund, Y., y Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1), 119–139.
- Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, 1189–1232.
- Geurts, P., Ernst, D., y Wehenkel, L. (2006). Extremely randomized trees. *Machine learning*, 63(1), 3–42.
- Holt, C. (1957). Forecasting trends and seasonals by exponentially weighted averages, vol. 52. *Carnegie Institute of Technology, Pittsburgh, Pa, USA*.

- Holt, C. C. (2004). Forecasting seasonals and trends by exponentially weighted moving averages. *International journal of forecasting*, 20(1), 5–10.
- Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8), 2554–2558.
- Kingma, D. P., y Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Mitchell, T. M., y cols. (1997). Machine learning.
- Naylor, B., Amanatides, J., y Thibault, W. (1990). Merging bsp trees yields polyhedral set operations. *ACM Siggraph Computer Graphics*, 24(4), 115–124.
- Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6), 386.
- Sarker, B. R., Jamal, A., y Wang, S. (2000). Supply chain models for perishable products under inflation and permissible delay in payment. *Computers & Operations Research*, 27(1), 59–75.
- Sathyanarayana, S. (2014, 07). A gentle introduction to backpropagation. *Numeric Insight, Inc Whitepaper*.
- Surowiecki, J. (2005). *The wisdom of crowds*. Anchor.
- Taylor, J. W. (2003). Short-term electricity demand forecasting using double seasonal exponential smoothing. *Journal of the Operational Research Society*, 54(8), 799–805.
- Tukey, J. W., y cols. (1977). *Exploratory data analysis* (Vol. 2). Reading, Mass.
- Vandeput, N. (2021). *Data science for supply chain forecasting*. De Gruyter.
- Winters, P. R. (1960). Forecasting sales by exponentially weighted moving averages. *Management science*, 6(3), 324–342.
- Yiu, T. (2019). *Understanding random forest*. <https://towardsdatascience.com/understanding-random-forest-58381e0602d2>. ([Online; accessed 29-Mayo-2021])
-